Simultaneous Data Recovery in Image and Transform Domains

ZHOU JUNQI

A THESIS SUBMITTED FOR THE DEGREE OF MASTER OF SCIENCE

Supervisor: Zuowei Shen

Department of Mathematics

National University of Singapore

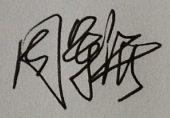
March, 2013

DECLARATION

I hereby declare that the thesis is my original work and it has been written by me in its entirety.

I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has not been submitted for any degree in any university previously.



Zhou Junqi

January 2013

Acknowledgments

I would like to acknowledge and present my heartful gratitude to my supervisor Prof. Zuowei Shen for his patience and constant guidance. Besides, I would like to thank Prof. Say Song Goh for his help.

Contents

Summary

This thesis addresses the problem of image recovery from partially given data in both the image and tight frame transform domains. Firstly, we consider a special case for the problem. In that case, the given data are the original image restricted on the support index set in the image domain and the canonical coefficients restricted on the support index set in the transform domain. Motivated by an uncertainty principle, a sufficient condition that ensures the exact recovery of an image is derived. The corresponding recovery algorithm is also given. Furthermore, we compare our algorithm with an existing reconstruction algorithm and see the similarity between them.

Then an analysis based model is proposed to handle situations in which exact recovery is impossible or unnecessary, such as when insufficient or only inaccurate data is available. An efficient iterative algorithm is obtained for the model by applying the split Bregman method. Several numerical examples are presented to demonstrate the potential of the algorithm.

List of Figures

Chapter 1

Introduction

1.1 Image Restoration in Image and Transform Domains

Image inpainting problem is an interesting and important inverse problem. It arises for example in restoring ancient drawings, in removing scratches in photos, and filling in the pixels of images when corrupted by noises. We need to find a solution for this inverse problem that is close to the given observed data. Furthermore, in this process, we are required to preserve the edges and some preferred regularities of the image.

In many problems in image processing, the data in the image domain and in the transform domain under certain transforms (such as the wavelet transform, discrete fourier transform, etc.) are both incomplete. In this thesis, we will focus on this problem.

We denote $\mathbb{R}^n$ to be the image domain by concatenating the columns of the image and $\mathbf{f} \in \mathbb{R}^n$ be the original image. In the image domain, only the data on the index set $\Lambda \subset \mathcal{N} = \{1, \ldots, n\}$ are given and we assume the given data is $\mathbf{x}$. In general, we have $\mathbf{P}_\Lambda \mathbf{f} = \mathbf{x}$ where $\mathbf{P}_\Lambda$ is the diagonal projective matrix defined

as

$$\mathbf{P}_\Lambda(i,j) = \begin{cases} 1 & \text{if} \quad i = j \in \Lambda \\ 0 & \text{otherwise} \end{cases}$$

Let $\mathbb{R}^m$ be the transform domain when we consider the transform operator $\mathbf{W}$ to be a $m \times n$ matrix. The data on the index set $\Gamma \subset \mathcal{M} := \{1, \ldots, m\}$ are given and we assume the given data is $\mathbf{y}$. Then in the transform domain, we have $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f} = \mathbf{y}$ and the projective matrix $\mathbf{P}_\Gamma$ is defined similar to $\mathbf{P}_\Lambda$.

Therefore, for the problem that contains missing data in both image and transform domains, we need to recover $\mathbf{f}$ or get an approximation of it which satisfies

$$\begin{cases} \mathbf{P}_\Lambda \mathbf{f} = \mathbf{x} \\ \mathbf{P}_\Gamma \mathbf{W}\mathbf{f} = \mathbf{y} \end{cases} \tag{1.1}$$

The problem (1.1) is an ill-posed inverse problem. It may have trivial solutions in some cases. For example, when $\Lambda = \mathcal{N}$ and $\Gamma = \emptyset$, then $\mathbf{f} = \mathbf{x}$ if $\mathbf{x}$ contains no noise, or it reduces to a denoising problem otherwise. The problem (1.1) can also have infinitely many solutions in some cases. For example, when $\Lambda \subset \mathcal{N}$ and $\Gamma = \emptyset$, one can choose any values to fill in the region $\mathcal{N} \backslash \Lambda$. In these cases, we need to impose some regularization conditions on the solution such that the chosen solution has certain smoothness requirements among all possible solutions. Yet in some other cases, the problem (1.1) may have no solution at all. For example, when the data set $\mathbf{y}$ falls out of the range of $\mathbf{P}_\Gamma \mathbf{W}$. This is possible, since the range of $\mathbf{W}$ is the orthogonal compliment of the kernel of $\mathbf{W}^T$ which is not empty when $\mathbf{W}$ is a redundant system. Even when $\mathbf{y}$ does fall inside of the range of $\mathbf{P}_\Gamma \mathbf{W}$, the given data on Λ may not be compatible with the given data on Γ and this results in (1.1) having no solution again. In these cases, we choose a solution $\mathbf{f}^*$ so that $\mathbf{P}_\Lambda \mathbf{f}^*$ is close to $\mathbf{x}$ in the image domain and $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}^*$ is close to $\mathbf{y}$ in the transform domain in some sense.

For the problem (1.1) ,the authors proposed the following iterative algorithm in [5]

$$\mathbf{f}_{k+1} = \mathbf{x} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{W}^T\mathbf{T}_\mu(\mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k) \tag{1.2}$$

where $\mathbf{T}_\mu$ is the soft thresholding operator

$$\mathbf{T}_\mu(\mathbf{y}) \ := (t_{\mu(1)}(\mathbf{y}(1)), \cdots, t_{\mu(i)}(\mathbf{y}(i)), \cdots, t_{\mu(m)}(\mathbf{y}(m)))$$

defined in [18] with

$$t_{\mu(i)}(\mathbf{y}(i)) \ := \begin{cases} 0 & \text{if } |\mathbf{y}(i)| \le \mu(i), \\ \mathbf{y}(i) - \operatorname{sgn}(\mathbf{y}(i))\mu(i) & \text{if } |\mathbf{y}(i)| \ge \mu(i). \end{cases}$$

We will give the details of this algorithm in Chapter two and show that the iteration generated by (1.2) converges to the variational model: Let $\mathbf{t}_k = \mathbf{T}_\mu(\mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k)$, then $\{\mathbf{t}_k\}_{k\ge 0}$ converges to $\mathbf{t}^*$ which is a minimizer of the following minimization problem

$$\min_{\{\mathbf{t}\in\mathbb{R}^m : \mathbf{P}_\Gamma\mathbf{t} = \mathbf{T}_\mu\mathbf{y}\}} \{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{W}^T\mathbf{t} - \mathbf{x}\|_2^2 + \frac{1}{2}\|(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{t}\|_2^2 + \|\operatorname{diag}(\mu)\mathbf{t}\|_1\}, \tag{1.3}$$

and the solution is given as $\mathbf{f}^* = \mathbf{W}^T\mathbf{t}^*$.

This model solves the problem in the transform domain. The first term penalizes the distance of the given data $\mathbf{x}$ to the solution $\mathbf{W}^T\mathbf{t}^*$. The second term penalizes the distance between the coefficients $\mathbf{t}$ and the canonical coefficients of the tight frame transform $\mathbf{W}$. Hence the second term is related to the smoothness of $\mathbf{f}^*$, since canonical coefficients of a transform is often linked to the smoothness of the underlying function. For example, some weighted norm of the canonical framelet coefficients is equivalent to some Besov norm of the underlying function (see for instance [26]). The third term is to ensure the sparsity of the transform coefficients, which in turn ensures the sharpness of the edges. Therefore when

the data $\mathbf{x}$ and $\mathbf{y}$ are arbitrarily given, and assume the the underlying solution has a good sparse approximation in transform domain, this model balances the approximation to the data fidelity and sparsity in the transform domain.

One special case for the above problem (1.1) is that the given data are $\mathbf{P}_\Lambda\mathbf{f}$ in the image domain and $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ in the transform domain respectively, i.e., $\mathbf{x} = \mathbf{P}_\Lambda\mathbf{f}$, $\mathbf{y} = \mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ in (1.1). For this case, we will prove in this thesis that if the transform $\mathbf{W}$ is tight frame transform, and the support index sets Λ in the image domain and Γ in the transform domain satisfy $\sum_{i\notin\Gamma}\sum_{j\notin\Lambda}|\mathbf{W}(i,j)|^2 < 1$ where $\mathbf{W}(i,j)$ is the (i,j)-th entry of the transform matrix $\mathbf{W}$, we can reconstruct the original data $\mathbf{f}$ exactly by applying the following iterative algorithm:

$$\mathbf{f}_{k+1} = \mathbf{P}_\Lambda\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{W}^T(\mathbf{P}_\Gamma\mathbf{W}\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k) \tag{1.4}$$

The above algorithm is essentially interpolating the given data in image domain and transform domain alternately. We can see in (1.4) that for an approximation $\mathbf{f}_k$ of the underlying solution $\mathbf{f}$, we firstly transform $\mathbf{f}_k$ into the transform domain and replace the data on Γ by the given data $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$. After that we transform it back to the image domain and replace the data on Λ by the given data $\mathbf{P}_\Lambda\mathbf{f}$. This gives the approximation $\mathbf{f}_{k+1}$ of $\mathbf{f}$ and then go on to the next iteration.

For the above special case, i.e., the given data is $\mathbf{P}_\Lambda\mathbf{f}$ in the image domain and $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ in the transform domain, the image restoration algorithm (1.2) becomes:

$$\mathbf{f}_{k+1} = \mathbf{P}_\Lambda\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{W}^T\mathbf{T}_\mu(\mathbf{P}_\Gamma\mathbf{W}\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k) \tag{1.5}$$

where $\mathbf{T}_\mu$ is the soft thresholding operator.

It is interesting to know that the two algorithm (1.4) and (1.5) are quite similar. The only difference between these two algorithms is that the denoising soft thresholding operator is applied in (1.5). This means that we may use (1.5) when the exact recovery condition does not hold or when the given data is contaminated

by noises.

1.2 Wavelets and Frames

Our approaches in this thesis are based on tight frame method, i.e., the transform operator $\mathbf{W}$ used in this thesis is tight frame transform. In this part we will give some preliminaries of tight framelets (see, e.g.,[6]). We firstly present the univariate framelets and the framelets for two variables can be constructed by tensor product of univariate framelets. The following part are mainly taken from [7, 11].

1.2.1 Framelets in $L_2(\mathbb{R})$

A system $X \in L_2(\mathbb{R})$ is called a tight frame of $L_2(\mathbb{R})$ if

$$\mathbf{f} = \sum_{\mathbf{x} \in X} \langle \mathbf{f}, \mathbf{x} \rangle \mathbf{x}, \quad \forall\, \mathbf{f} \in L_2(\mathbb{R}) \tag{1.6}$$

This is equivalent to

$$\|\mathbf{f}\|_2 = \sum_{\mathbf{x} \in X} \|\langle \mathbf{f}, \mathbf{x} \rangle\|_2^2, \quad \forall\, \mathbf{f} \in L_2(\mathbb{R}) \tag{1.7}$$

where $\langle \cdot, \cdot \rangle$ and $\| \cdot \|_2^2$ are the inner product and norm of $L_2(\mathbb{R})$. It is clear that an orthonormal basis is a tight frame system, since the identities (1.6) and (1.7) hold for arbitrary orthonormal basis in $L_2(\mathbb{R})$. Hence tight frames are generalization of orthonormal basis that bring in the redundancy which is often useful in applications such as denoising (see e.g. [14]). Recall that a wavelet (or affine) system $X(\Psi)$ is defined to be the collection of dilations and shifts of a finite set $\Psi \in L_2(\mathbb{R})$, i.e.,

$$X(\Psi) = \{2^{k/2}\psi(2^k x - j) : \psi \in \Psi, k, j \in \mathbb{Z}\}$$

and the elements in Ψ are called the generators. When $X(\Psi)$ is also a tight frame for $L_2(\mathbb{R})$, then $\psi \in \Psi$ are called (tight) framelets, following the terminology used in [17].

To construct compactly supported framelet systems, one starts with a compactly supported refinable function $\phi \in L_2(\mathbb{R})$ with a refinement mask (low-pass filter) ζ_ϕ such that ϕ satisfies the refinement equation: $\hat{\phi}(2\cdot) = \zeta_\phi \hat{\phi}$. Here $\hat{\phi}$ is the Fourier transform of ϕ, and ζ_ϕ is a trigonometric polynomial with $\zeta_\phi(0) = 1$. A multiresolution analysis (MRA) from this given refinable function can be formed, see [2, 28]. The compactly supported framelets Ψ are defined in the Fourier domain by $\hat{\psi}(2\cdot) = \zeta_\psi \hat{\phi}$ for some trigonometric polynomials ζ_ψ, $\psi \in \Psi$. The unitary extension principle (UEP) of [30] asserts that the system $X(\Psi)$ generated by the finite set Ψ forms a tight frame in $L_2(\mathbb{R})$ provided that the masks ζ_ϕ and $\{\zeta_\psi\}_{\psi\in\Psi}$ satisfy:

$$\zeta_\phi \overline{\zeta_\phi(\omega + \gamma\pi)} + \sum_{\psi\in\Psi} \zeta_\psi \overline{\zeta_\psi(\omega + \gamma\pi)} = \delta_{\gamma,0}, \quad \gamma = 0, 1 \tag{1.8}$$

for almost all $\omega \in \mathbb{R}$. The sequences of Fourier coefficients of ζ_ψ, as well as ζ_ψ itself, are called framelet masks or high-pass filters. The construction of framelets Ψ essentially is to design framelet masks $\{\zeta_\psi\}_{\psi\in\Phi}$ for a given refinement mask ζ_ϕ such that (1.8) holds. For a given ϕ with refinement mask ζ_ϕ, as shown in [15, 17], it is easy to construct $\zeta_\psi, \psi \in \Psi$ whenever ζ_ϕ satisfies

$$|\zeta_\phi|^2 + |\zeta_\phi(\cdot + \pi)|^2 \leq 1$$

Furthermore, the framelets can be constructed to be symmetric as long as ϕ is symmetric. In particular, one can construct tight framelet systems from B-splines. Here, we give two examples.

The first example is derived from piecewise linear B-spline whose refinement

mask is $h_0 = \frac{1}{4}[1, 2, 1]$. The two corresponding framelet masks are

$$h_1 = \frac{\sqrt{2}}{4}[1, 0, -1], h_2 = \frac{1}{4}[-1, 2, -1]$$

The second example is derived from piecewise cubic B-spline whose refinement mask is $h_0 = \frac{1}{16}[1, 4, 6, 4, 1]$. The four framelet masks are

$$h_1 = \frac{1}{8}[1, 2, 0, -2, -1]; h_2 = \frac{\sqrt{6}}{16}[-1, 0, 2; 0, -1]; h_3 = \frac{1}{8}[-1, 2, 0, -1, 1]; h_4 = \frac{1}{16}[1, -4, 6, -4, 1]$$

Construction of tight framelets from B-splines of high orders can be found in [30]. The refinement and framelet masks can be used to derive fast decomposition and reconstruction algorithms similar to the orthonormal wavelet case. Interested readers can refer [9, 30] for more details.

1.2.2 Frames in $\mathbb{R}^n$

Since images are finite dimensional, we describe briefly here how to convert the framelet decomposition and reconstruction to finite dimension frames. Let $\mathbf{W}$ be a m-by-n ($n \leq m$) matrix whose rows are vectors in $\mathbb{R}^n$. The system, denoted by $\mathbf{W}$ again, consisting of all the rows of $\mathbf{W}$, is a tight frame for $\mathbb{R}^n$ if for any vector $\mathbf{f} \in \mathbb{R}^n$, we have

$$\|\mathbf{f}\|_2^2 = \sum_{\mathbf{x} \in \mathbf{W}} \|\langle \mathbf{f}, \mathbf{x} \rangle\|_2^2$$

Note that the above equation is equivalent to the perfect reconstruction formula $\mathbf{f} = \sum_{\mathbf{x} \in \mathbf{W}} \langle \mathbf{f}, \mathbf{x} \rangle \mathbf{x}$. The matrix $\mathbf{W}$ is called the analysis (or decomposition) operator, and its adjoint $\mathbf{W}^T$ is called the synthesis (or reconstruction) operator. The perfect reconstruction formula can be rewritten as $\mathbf{f} = \mathbf{W}^T \mathbf{W} \mathbf{f}$. Hence $\mathbf{W}$ is a tight frame if and only if $\mathbf{W}^T \mathbf{W} = \mathbf{I}$. Unlike the orthonormal basis, we emphasize that $\mathbf{W}\mathbf{W}^T \neq \mathbf{I}$ in general. Or else the system of the rows of $\mathbf{W}$ form an orthonormal basis. The basic assumption for tight frame based image restoration

is that the real images can be sparse represented by tight frame. This "sparse approximation" is the key in many problems in applications.

In the following, we derive the tight frame system $\mathbf{W}$ from the given masks $\{h_k\}_{0\le k\le 2m}$. Let h be a filter with length $2m+1$, i.e.,

$$h = [h(-m), h(-m+1), \cdots, h(-1), h(0), \cdots, h(m-1), h(m)]$$

If Neumann (symmetric) boundary condition is used, then the matrices representation of h will be an $n \times n$ matrix H given by

$$H(i,j) = \begin{cases} h(i-j) + h(i+j-1), & \text{if } i+j \le m+1 \\ h(i-j) + h(-1-(2n-i-j)), & \text{if } i+j \ge 2n-m+1 \\ h(i-j), & \text{otherwise} \end{cases} \tag{1.9}$$

When the filter h is symmetric, the resulting matrix H is a Toeplitz-plus-Toeplitz and its spectra can be computed easily (see, e.g., [12]). We note that Neumann boundary conditions usually produce restored images having less artifacts near the boundary, see [10, 12] for instances.

Next we define the matrix $\mathcal{L}_k$ and $\mathcal{H}_k$:

$$\mathcal{L}_k = H_0^{(k)} H_0^{(k-1)} \cdots H_0^{(1)} \quad \text{and} \quad \mathcal{H}_k = \begin{bmatrix} H_1^{(k)} \mathcal{L}_{k-1} \\ \vdots \\ H_{2m}^{(k)} \mathcal{L}_{k-1} \end{bmatrix}, \quad k > 0 \tag{1.10}$$

where $H_k^{(l)}$ is the matrix representation of the filters formed from h_k by inserting $2^{l-1}-1$ zeros between every two adjacent components of h_k. The multi-level decomposition operator $\mathbf{W}$ up to level L induced from the spline tight framelets

is given as follows

$$\mathbf{W} := \begin{bmatrix} \mathcal{L}_L \\ \mathcal{H}_L \\ \mathcal{H}_{L-1} \\ \vdots \\ \mathcal{H}_1 \end{bmatrix} \equiv \begin{bmatrix} \mathbf{W}_0 \\ \mathbf{W}_1 \end{bmatrix}$$

where $\mathbf{W}_0 = \mathcal{L}_L$ and $\mathbf{W}_1$ consists of the remaining blocks of $\mathbf{W}$.

The unitary extension principle asserts that

$$\mathbf{W}^T\mathbf{W} = \mathbf{W}_0^T\mathbf{W}_0 + \mathbf{W}_1^T\mathbf{W}_1 = \mathbf{I}$$

Hence $\mathbf{W}$ is a tight frame in $\mathbb{R}^n$.

So far we have only considered tight framelet systems in 1-D. Since images are 2-D objects, when we handle images, we use tensor product tight framelet system generated by the corresponding univariate tight framelet system. Let $H_k^{(l)}$,$0 \leq k \leq 2m$ and $1 \leq l < L$, be matrices defined in (1.10). Define

$$G_{(2m+1)i+j}^{(l)} := H_i^{(l)} \otimes H_j^{(l)}, \quad 0 \leq i, j \leq 2m$$

It is obvious that we have

$$\sum_{k=1}^{(2m+1)^2-1} (G_k^{(l)})^T G_k^{(l)} = I$$

Then we define

$$\mathcal{L}_k := G_0^{(k)} G_0^{(k-1)} \cdots G_0^{(1)} \quad \text{and} \quad \mathcal{H}_k = \begin{bmatrix} G_1^{(k)} \mathcal{L}_{k-1} \\ \vdots \\ G_{(2m+1)^2}^{(k)} \mathcal{L}_{k-1} \end{bmatrix} \quad k > 0$$

With these notations, we can form the matrix similar to the 1-D case

$$\mathbf{W} := \begin{bmatrix} \mathcal{L}_L \\ \mathcal{H}_L \\ \mathcal{H}_{L-1} \\ \vdots \\ \mathcal{H}_1 \end{bmatrix}$$

and $\mathbf{W}$ is a tight frame from the unitary extension principle.

1.3 Motivation, Contribution and Structure

As stated in the previous part, for some image $\mathbf{f} \in \mathbb{R}^n$, when $\mathbf{W}$ is tight frame transform, the given data are $\mathbf{P}_\Lambda \mathbf{f}$ in the image domain and $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in the transform domain ($\mathbf{x} = \mathbf{P}_\Lambda \mathbf{f}$ and $\mathbf{y} = \mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in (1.1)), we can exactly recover the original data $\mathbf{f}$ by algorithm (1.4) when the sufficient condition $\sum_{i \notin \Gamma} \sum_{j \notin \Lambda} |\mathbf{W}(i,j)|^2 < 1$ holds. However, when the exact recovery condition does not hold or the data $\mathbf{x}$ and $\mathbf{y}$ are arbitrarily given, we can get an approximate solution from the algorithm (1.2) derived by solving the model (1.3) which is a balanced approach model. While the algorithm (1.2) is efficient, it may not very much closely related to $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in the transform domain, when the given data is closely related to $\mathbf{P}_\Lambda \mathbf{f}$ and $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$. It is more proper to have a model whose approximation term in the transform domain is reflected by $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$. Note that, since $\mathbf{W}$ is redundant, for given $\mathbf{f}$ there are infinitely many $\mathbf{t}$ such that $\mathbf{f} = \mathbf{W}^T \mathbf{t}$. In the frame literature $\mathbf{W}\mathbf{f}$ is called canonical coefficients of the frame transform of $\mathbf{f}$. In many cases, the sparsity assumption is also imposed on the canonical coefficients which is also reflected by the regularity term in the model. Altogether, we propose the following analysis based model when the exact recovery is impossible or unnecessary and when approximation and regularity of $\mathbf{W}\mathbf{f}$ are desirable: The solution is a

minimizer of the following minimization problem

$$\min_{\mathbf{f}\in\mathbb{R}^n}\{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{W}\mathbf{f}\|_1\} \tag{1.11}$$

where $\nu > 0$ is a weighted parameter and μ is a positively weighted vector. The first term penalizes the distance of $\mathbf{P}_\Lambda\mathbf{f}$ to the given data $\mathbf{x}$ in the image domain. The second term penalizes the distance of $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ to the given data $\mathbf{y}$ in the transform domain. Thus the first two terms in (1.11) penalize the distance of the given data to the solution in both image and transform domains. The third term guarantees the regularity and sparsity of the underlying solution. We will derive an efficient iterative algorithm for the model (1.11) by using split Bregman method (see [8]).

The rest of the thesis is organized as follows. In chapter two, we will introduce the balanced approach algorithm in details. Furthermore, to accelerate the convergence rate of the algorithm, accelerated proximal gradient(APG) algorithm for the balanced approach algorithm is proposed by applying the idea in [31]. The corresponding convergence rate of these two algorithms are also given. In chapter three, we focus on a special case for the problem (1.1), i.e., $\mathbf{x} = \mathbf{P}_\Lambda\mathbf{f}$ and $\mathbf{y} = \mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ in (1.1). A sufficient condition which enables $\mathbf{f}$ can be exactly recovered is given and the reconstruction algorithm is also proposed. In chapter four, for the case that the exact recovery condition does not hold or the data $\mathbf{x}$ and $\mathbf{y}$ are arbitrarily given, we proposed an analysis based model for (1.1) and derive our algorithm by using split Bregman method. Some implementations of our algorithm are presented.

Chapter 2

Balanced Approach Image Restoration

Image inpainting is to recover data by interpolation. There are many interpolation schemes available, e.g., spline interpolation, but majority of them are only good for smooth functions. Images are either piecewise smooth function or texture which do not have the required globe smoothness to provide a good approximation of underlying solutions. The major challenge in image inpainting is to keep the features, e.g., edges of images, which many of those available interpolation algorithms cannot preserve. Furthermore, since images are usually contaminated by noises, the algorithms should have a build in denoising component. In this part, we will introduce the balanced approach image restoration.

2.1 Balanced Approach Image Restoration

For arbitrary given data $\mathbf{x}$ supported on $\Lambda \subset \mathcal{N}$ and $\mathbf{y}$ supported on $\Gamma \subset \mathcal{M}$, we want to recover the original image $\mathbf{f}$ which satisfies

$$\begin{cases} \mathbf{P}_\Lambda \mathbf{f} = \mathbf{x} \\ \mathbf{P}_\Gamma \mathbf{W} \mathbf{f} = \mathbf{y} \end{cases}$$

The simple idea of the balanced approach for frame based image restoration comes as follows: one may use any simple interpolation scheme to interpolate the given data in both image and transform domains that leads to an inpainted image. The edges might be blurred in this inpainted image. One of the simplest ways to sharpen the image is to threw out small coefficients under a tight wavelet frame transform. This deletion of small wavelet frame coefficients not only sharpens edges but also removes noises. When it is reconstructed back to image domain, it will not interpolate the data anymore, the simplest way to make it interpolate the given data is to put the given data back. One may iterate this process till convergence.

To be precise, the authors in [5] proposed the iterative algorithm (1.2) for the problem (1.1):

$$\mathbf{f}_{k+1} = \mathbf{x} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{W}^T \mathbf{T}_\mu (\mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k) \tag{2.1}$$

where $\mathbf{T}_\mu$ is the soft thresholding operator.

From $\mathbf{f}_k$ to $\mathbf{f}_{k+1}$, we first transform $\mathbf{f}_k$ to the transform domain to get the transform coefficients $\mathbf{W}\mathbf{f}_k$. Then we replace the data on Γ by the given data $\mathbf{y}$. After that, we apply the soft thresholding operator $\mathbf{T}_\mu$ on the coefficients $\mathbf{y}+(\mathbf{I}-\mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k$ to perturb the transform coefficients and to remove possible noise. Finally, the modified coefficients are transformed back to the image domain, and

the data on Λ is replaced by the given data $\mathbf{x}$. This gives the next approximation $\mathbf{f}_{k+1}$.

By letting $\mathbf{t}_k = \mathbf{T}_\mu(\mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k)$, the iteration (2.1) can be written as

$$\begin{cases} \mathbf{t}_k = \mathbf{T}_\mu(\mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k) \\ \mathbf{f}_{k+1} = \mathbf{x} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{W}^T\mathbf{t}_k \end{cases} \tag{2.2}$$

The authors in [5] prove that the limit $\mathbf{t}^*$ of $\mathbf{t}_k$ is a minimizer of

$$\min_{\{\mathbf{t}\in\mathbb{R}^m : \mathbf{P}_\Gamma\mathbf{t} = \mathbf{T}_\mu\mathbf{y}\}} \{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{W}^T\mathbf{t} - \mathbf{x}\|_2^2 + \frac{1}{2}\|(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{t}\|_2^2 + \|\mathrm{diag}(\mu)\mathbf{t}\|_1\}, \tag{2.3}$$

and the solution is given as $\mathbf{f}^* = \mathbf{W}^T\mathbf{t}^*$.

The idea of the convergence proof is that the sequence $\{\mathbf{t}_k\}_{k\geq 0}$ in (2.2) can be written as a proximal forward-backward splitting iteration of (2.3). If we define the set $\mathcal{I}$ and the indicator function $\iota_{\mathcal{I}}$ as

$$\mathcal{I} := \{\mathbf{t} \in \mathbb{R}^m : \mathbf{P}_\Gamma\mathbf{t} = \mathbf{T}_\mu\mathbf{y}\}$$

and

$$\iota_{\mathcal{I}}(\mathbf{t}) = \begin{cases} 0, \text{if } \mathbf{t} \in \mathcal{I} \\ +\infty, \text{otherwise} \end{cases}$$

Then, the balanced based approach model (2.3) can be written as

$$\min_{\mathbf{t}\in\mathbb{R}^m} \{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{W}^T\mathbf{t} - \mathbf{x}\|_2^2 + \frac{1}{2}\|(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{t}\|_2^2 + \xi(\mathbf{t})\} \tag{2.4}$$

where $\xi(\mathbf{t}) := \|\mathrm{diag}(\mu)\mathbf{t}\|_1 + \iota_{\mathcal{I}}(\mathbf{t})$.

By letting

$$\mathbf{F}_1(\mathbf{t}) = \xi(\mathbf{t}),\ \mathbf{F}_2(\mathbf{t}) = \frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{W}^T\mathbf{t} - \mathbf{x}\|_2^2 + \frac{1}{2}\|(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{t}\|_2^2 \tag{2.5}$$

the authors show in [5] that the reconstruction algorithm (2.2) is equivalent to the proximal forward-backward splitting (PFBS) iteration for (2.4):

$$\mathbf{t}_{k+1} = \text{prox}_{\mathbf{F}_1}\{\mathbf{t}_k - \nabla \mathbf{F}_2(\mathbf{t}_k)\}$$

where $\text{prox}_{\varphi}(\mathbf{r})$ is the proximal operator of φ defined by

$$\text{prox}_{\varphi}(\mathbf{r}) = \arg\min_{\mathbf{t}\in\mathbb{R}^m}\{\frac{1}{2}\|\mathbf{r}-\mathbf{t}\|_2^2 + \varphi(\mathbf{t})\}$$

Note that if the $\mathbf{F}_1(\mathbf{t}) = \xi(\mathbf{t})$ is defined as above, then the proximal operator of F_1 is

$$\text{prox}_{\mathbf{F}_1}(\mathbf{r}) = \arg\min_{\mathbf{t}\in\mathbb{R}^m}\{\frac{1}{2}\|\mathbf{r}-\mathbf{t}\|_2^2 + \|\text{diag}(\mu)\mathbf{t}\|_1 + \iota_{\mathcal{I}}(\mathbf{t}\} = \mathbf{T}_{\mu}(\mathbf{y} + (\mathbf{I}-\mathbf{P}_{\Gamma})\mathbf{r})$$

Then, the balanced approach reconstruction algorithm (2.2) can be written in the following form: Set initial guesses $\mathbf{t}_0 = \mathbf{t}_{-1} \in \mathbb{R}^m$, the iteration (2.2) for the balanced approach is

$$\begin{cases} \mathbf{g}_k = \mathbf{t}_k - \nabla \mathbf{F}_2(\mathbf{t}_k) \\ \mathbf{t}_{k+1} = \text{prox}_{\mathbf{F}_1}(\mathbf{g}_k) \end{cases} \tag{2.6}$$

The convergence of the proximal forward-backward splitting is guaranteed by the following theorem in [13]

Theorem 2.1.1. *Consider the minimization problem*

$$\min_{\mathbf{t}\in\mathbb{R}^m} \mathbf{F}_1(\mathbf{t}) + \mathbf{F}_2(\mathbf{t}) \tag{2.7}$$

where $\mathbf{F}_1 : \mathbb{R}^m \to \mathbb{R}$ *is a proper, convex, lower semi-continuous function and* $\mathbf{F}_2 : \mathbb{R}^m \to \mathbb{R}$ *is a convex, differentiable function with an L-Lipschitz continuous gradient. Assume a minimizer of* (2.7) *exists. Then for any initial guess* $\mathbf{t}_0$ *, the*

iteration (called the proximal forward-backward splitting):

$$\mathbf{t}_{k+1} = \text{prox}_{\mathbf{F}_1/L}(\mathbf{t}_k - \nabla \mathbf{F}_2(\mathbf{t}_k)/L) \tag{2.8}$$

converges to the minimizer of $\mathbf{F}_1(\mathbf{t}) + \mathbf{F}_2(\mathbf{t})$.

We will not prove this Theorem since the conclusion of this Theorem is included in Theorem 2.1.2 below. It is easy to verify that $\mathbf{F}_1(\mathbf{t})$, $\mathbf{F}_2(\mathbf{t})$ defined in (2.5) satisfy the conditions in Theorem 2.1.1 and $F_2(\mathbf{t})$ is 1-Lipschitz (see,e.g.,[16]). Thus the iteration $\{\mathbf{t}_k\}_{k\geq 0}$ in (2.6) converges to a minimization of model (2.4), and hence model (2.3).

In [16], the authors considered the image inpainting problem ($\Gamma = \emptyset$ in (1.1)) and the convergence rate is given. The PFBS algorithm can still be written as (2.6) with $\mathbf{F}_1(\mathbf{t}) = \|\text{diag}(\mu)\mathbf{t}\|_1$. For the two domain image restoration problem (1.1), the PFBS algorithm is (2.6) with $\mathbf{F}_1(\mathbf{t}) = \|\text{diag}(\mu)\mathbf{t}\|_1 + \iota_{\mathcal{I}}(\mathbf{t})$. With different definition of $\mathbf{F}_1$, we still have a similar result for the convergence rate. The arguments is quite similar and we will give a proof in order to make the thesis self-contained.

For notational convenience we denote $\mathbf{F}(\mathbf{t}) = \mathbf{F}_1(\mathbf{t}) + \mathbf{F}_2(\mathbf{t})$ and

$$l_{\mathbf{F}}(\alpha;\beta) = \mathbf{F}_2(\beta) + \langle \nabla \mathbf{F}_2(\beta), \alpha - \beta\rangle + \mathbf{F}_1(\alpha)$$

where the sum of the first two terms is the linear approximation of $\mathbf{F}_2$ at β. Since $\mathbf{F}_2$ has an L-Lipschitz continuous gradient and is convex, we have the following inequality

$$\mathbf{F}(\alpha) - \frac{L}{2}\|\alpha - \beta\|_2^2 \leq l_{\mathbf{F}}(\alpha;\beta) \tag{2.9}$$

We have the following theorem which reveals the convergence rate of the PFBS iteration (2.8)

Theorem 2.1.2. *Consider the minimization problem*

$$\min_{\mathbf{t}\in\mathbb{R}^m} \mathbf{F}_1(\mathbf{t}) + \mathbf{F}_2(\mathbf{t}) \tag{2.10}$$

where $\mathbf{F}_1(\mathbf{t}) = \|\mathrm{diag}(\mu)\mathbf{t}\|_1 + \iota_{\mathcal{I}}(\mathbf{t})$ *and* $F_2 : \mathbb{R}^m \to \mathbb{R}$ *is a convex, differentiable function with an L-Lipschitz continuous gradient. Let* $\mathbf{F} := \mathbf{F}_1 + \mathbf{F}_2$ *and denote by* $\mathbf{t}^*$ *a solution of* (2.10)*. Then the sequence* $\{\mathbf{t}_k\}_{k\geq 0}$ *generated by the iteration* (2.8) *satisfies*

$$\mathbf{F}(\mathbf{t}_k) - \mathbf{F}(\mathbf{t}^*) \leq \frac{L\|\mathbf{t}^* - \mathbf{t}_0\|_2^2}{2k} \tag{2.11}$$

As a consequence, for given $\epsilon > 0$*, we have*

$$\mathbf{F}(\mathbf{t}_k) - \mathbf{F}(\mathbf{t}^*) \leq \epsilon, \quad \text{whenever} \quad k \geq \frac{L(C + \|\mathbf{t}_0\|_2)^2}{2\epsilon} \tag{2.12}$$

where C *is a constant that satisfies* $\|\mathbf{t}^*\|_1 \leq C$*. When* (2.10) *has a unique solution* $\mathbf{t}^*$*, we have*

$$\lim_{k\to\infty} \|\mathbf{t}_k - \mathbf{t}^*\|_2^2 = 0$$

First, we recall the following result on convergence of minimizing sequences which is taken from [8]

Proposition 2.1.1. *Let* $\mathbf{F}(\mathbf{t})$ *be a convex function defined on* $\mathbb{R}^m$ *and nowhere assumes the values* $\pm\infty$*. Suppose* $\mathbf{F}$ *has a unique minimizer* $\mathbf{t}^* \in \mathbb{R}^m$*. Then any minimizing sequence* $\{\mathbf{t}_k\}_{k\geq 0}$*, i.e.,* $\mathbf{F}(\mathbf{t}_k) \to \mathbf{F}(\mathbf{t}^*)$ *as* $k \to +\infty$*, converges to* $\mathbf{t}^*$ *in any Euclidean norm of* $\mathbb{R}^m$*.*

Now we can prove Theorem 2.1.2

Proof of Thm. 2.1.2. For $k \geq 1$, we firstly show that

$$\mathbf{t}_{k+1} \in \arg\min_{\mathbf{t}\in\mathbb{R}^m} \{l_{\mathbf{F}}(\mathbf{t}; \mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}\rangle\} \tag{2.13}$$

By letting $\mathbf{g}_k = \mathbf{t}_k - \nabla \mathbf{F}_2(\mathbf{t}_k)/L$, it is easy to see that (2.13) is equivalent to

$$\mathbf{t}_{k+1} \in \arg\min_{\mathbf{t}\in\mathbb{R}^m}\{\langle \mathbf{t}_{k+1} - \mathbf{g}_k, \mathbf{t}\rangle + \mathbf{F}_1(\mathbf{t})/L\} \tag{2.14}$$

Since we have

$$\mathbf{t}_{k+1} = \text{prox}_{\mathbf{F}_1/L}(\mathbf{g}_k) = \arg\min_{\mathbf{t}\in\mathbb{R}^m}\{\|\mathbf{t} - \mathbf{g}_k\|_2^2 + \mathbf{F}_1(\mathbf{t})/L\}$$

we have

$$0 \in \mathbf{t}_{k+1} - \mathbf{g}_k + \partial \mathbf{F}_1(\mathbf{t}_{k+1})/L$$

which implies (2.14) and hence (2.13). From (2.13), we now have

$$l_{\mathbf{F}}(\mathbf{t}_{k+1}; \mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}_{k+1}\rangle \leq l_{\mathbf{F}}(\mathbf{t}^*; \mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}^*\rangle \tag{2.15}$$

Letting $\alpha = \mathbf{t}_{k+1}$ and $\beta = \mathbf{t}_k$ in (2.9), we get

$$\mathbf{F}(\mathbf{t}_{k+1}) \leq l_{\mathbf{F}}(\mathbf{t}_{k+1}; \mathbf{t}_k) + \frac{L}{2}\|\mathbf{t}_{k+1} - \mathbf{t}_k\|_2^2 \tag{2.16}$$

Applying (2.15) to (2.16), we have

$$\begin{aligned}
\mathbf{F}(\mathbf{t}_{k+1}) &\leq l_{\mathbf{F}}(\mathbf{t}^*; \mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}^* - \mathbf{t}_{k+1}\rangle + \frac{L}{2}\|\mathbf{t}_{k+1} - \mathbf{t}_k\|_2^2 \\
&= l_{\mathbf{F}}(\mathbf{t}^*; \mathbf{t}_k) + \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_k\|_2^2 - \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_{k+1}\|_2^2 \\
&\leq \mathbf{F}(\mathbf{t}^*) + \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_k\|_2^2 - \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_{k+1}\|_2^2
\end{aligned}$$

where the last inequality follows from the definition of $l_{\mathbf{F}}$ and the convexity of $\mathbf{F}_2$. Then we will have the following inequality

$$\mathbf{F}(\mathbf{t}_{k+1}) - \mathbf{F}(\mathbf{t}^*) \leq \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_k\|_2^2 - \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_{k+1}\|_2^2$$

Telescoping on the above inequality, we will get

$$\sum_{j=1}^{k+1} \mathbf{F}(\mathbf{t}_j) - (k+1)\mathbf{F}(\mathbf{t}^*) \leq \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_0\|_2^2 \tag{2.17}$$

By using (2.13) again, we have

$$\begin{aligned} l_{\mathbf{F}}(\mathbf{t}_{k+1}; \mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}_{k+1}\rangle &\leq l_{\mathbf{F}}(\mathbf{t}_k; \mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}_k\rangle \\ &= \mathbf{F}(\mathbf{t}_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \mathbf{t}_k\rangle \end{aligned}$$

By applying the above inequality to (2.16), we get

$$\mathbf{F}(\mathbf{t}_{k+1}) - \mathbf{F}(\mathbf{t}_k) \leq -\frac{L}{2}\|\mathbf{t}_{k+1} - \mathbf{t}_k\|_2^2$$

Multiplying k on both sides of the above inequality and then telescoping, we have

$$(k+1)\mathbf{F}(\mathbf{t}_{k+1}) - \sum_{j=1}^{k+1} \mathbf{F}(\mathbf{t}_j) = k\mathbf{F}(\mathbf{t}_{k+1}) - \sum_{j=1}^{k} \mathbf{F}(\mathbf{t}_j) \leq -\frac{L}{2}\sum_{j=1}^{k} j\|\mathbf{t}_{j+1} - \mathbf{t}_j\|_2^2 \tag{2.18}$$

Combining (2.17) and (2.18), we have

$$(k+1)(\mathbf{F}(\mathbf{t}_{k+1}) - \mathbf{F}(\mathbf{t}^*)) \leq \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_0\|_2^2$$

and thus (2.11) holds. In addition, by applying the triangle inequality

$$\|\mathbf{t}^* - \mathbf{t}_0\|_2 \leq \|\mathbf{t}^*\|_2 + \|\mathbf{t}_0\|_2 \leq \|\mathbf{t}^*\|_1 + \|\mathbf{t}_0\|_2$$

we obtain (2.12).

The conclusion that $\mathbf{t}_k \to \mathbf{t}^*$ when $\mathbf{t}^*$ is the unique minimizer of $\mathbf{F}$ follows directly from Proposition 2.1.1. ■

It is obvious by Theorem 2.1.2 that it requires $O(L/\epsilon)$ iterations to get an ϵ-

optimal solution. In the next section, we will introduce an acceleration algorithm for the PFBS algorithm.

2.2 Accelerated Proximal Gradient Method for Framed Based Image Restoration

As stated in the previous section, the proximal forward-backward splitting algorithm generates an ϵ-optimal solution in $O(L/\epsilon)$ iterations, which is reasonably efficient. However, in practice, faster algorithms are always desired. Therefore, one always wishes to reduce the total number of iterations to get an satisfactory solution. In [31],the authors adapt the accelerated proximal gradient (APG) algorithm to solve the l_1-regularized linear least squares problem in the balanced approach in frame based image restoration. We will follow this idea to derive the APG for (2.8) with incomplete data in both image and transform domains. The APG algorithm of [32] is obtained by adjusting the $\mathbf{g}_k$ step in the proximal forward-backward splitting algorithm. This idea has already appeared in [1, 34]. Next, we describe the APG algorithm for (2.8): Set initial guesses $\mathbf{t}_0 = \mathbf{t}_{-1} \in \mathbb{R}^m$,$s_0 = 1$,and $s_{-1} = 0$ and generate $\mathbf{t}_k$ by

$$\begin{cases} \beta_k = \mathbf{t}_k + \frac{s_{k-1}-1}{s_k}(\mathbf{t}_k - \mathbf{t}_{k-1}) \\ \mathbf{g}_k = \beta_k - \nabla F_2(\beta_k)/L \\ \mathbf{t}_{k+1} = \text{prox}_{F_1/L}(\mathbf{g}_k) \\ s_{k+1} = \frac{1+\sqrt{1+4s_k^2}}{2} \end{cases} \tag{2.19}$$

The convergence rate of the APG algorithm (2.19) is revealed by the following theorem which is similar to the Theorem 4.5 in [16]

Theorem 2.2.1. *Consider the minimization problem*

$$\min_{\mathbf{t}\in\mathbb{R}^m} \mathbf{F}_1(\mathbf{t}) + \mathbf{F}_2(\mathbf{t}) \tag{2.20}$$

where $\mathbf{F}_1(\mathbf{t}) = \|\mathrm{diag}(\mu)\mathbf{t}\|_1 + \iota_{\mathcal{I}}(\mathbf{t})$ *and* $\mathbf{F}_2 : \mathbb{R}^m \to \mathbb{R}$ *is a convex, differentiable function with an L-Lipschitz continuous gradient. Let* $\mathbf{F} := \mathbf{F}_1 + \mathbf{F}_2$ *and* $\{\mathbf{t}_k\}$, $\{\beta_k\}$, *and* $\{s_k\}$ *be the sequences generated by Algorithm* (2.19). *Then for any* $k \geq 1$ *and any optimal solution* $\mathbf{t}^*$ *to the minimization problem* (2.20) *with* $0 \leq k < \infty$, *we have*

$$\mathbf{F}(\mathbf{t}_k) - \mathbf{F}(\mathbf{t}^*) \leq \frac{L\|\mathbf{t}^* - \mathbf{t}_0\|_2^2}{2(k+1)^2} \tag{2.21}$$

Hence

$$\mathbf{F}(\mathbf{t}_k) - \mathbf{F}(\mathbf{t}^*) \leq \epsilon, \quad \text{whenever } k \geq \sqrt{\frac{L}{2\epsilon}}(\|\mathbf{t}_0\|_2 + C) - 1 \tag{2.22}$$

where C *is a constant satisfies* $\|\mathbf{t}^*\|_1 \leq C$. *Furthermore, if* $\mathbf{t}^*$ *is the unique minimizer of* $F(\mathbf{t})$, *then* $\mathbf{t}_k \to \mathbf{t}^*$ *as* $k \to \infty$.

Proof. For $k \geq 1$ and any optimal solution $\mathbf{t}^*$, let $\tilde{\mathbf{t}} = \frac{\mathbf{t}^* + (s_k - 1)\mathbf{t}_k}{s_k}$. We first show that

$$\mathbf{t}_{k+1} \in \arg\min_{\mathbf{t} \in \mathbb{R}^m} \{l_{\mathbf{F}}(\mathbf{t}; \beta_k) + L\langle \mathbf{t}_{k+1} - \beta_k, \mathbf{t}\rangle\} \tag{2.23}$$

which is equivalent to

$$\mathbf{t}_{k+1} \in \arg\min_{\mathbf{t} \in \mathbb{R}^m} \{\langle \mathbf{t}_{k+1} - \mathbf{g}_k, \mathbf{t}\rangle + \mathbf{F}_1(\mathbf{t})/L\} \tag{2.24}$$

Since we have

$$\mathbf{t}_{k+1} = \mathrm{prox}_{\mathbf{F}_1/L}(\mathbf{g}_k) = \arg\min_{\mathbf{t} \in \mathbb{R}^m} \{\|\mathbf{t} - \mathbf{g}_k\|_2^2 + \mathbf{F}_1(\mathbf{t})/L\}$$

we can get

$$0 \in \mathbf{t}_{k+1} - \mathbf{g}_k + \partial \mathbf{F}_1(\mathbf{t}_{k+1})/L$$

which implies (2.24) and hence (2.23). From (2.23), we now have

$$l_{\mathbf{F}}(\mathbf{t}_{k+1}; \beta_k) + L\langle \mathbf{t}_{k+1} - \beta_k, \mathbf{t}_{k+1}\rangle \leq l_{\mathbf{F}}(\tilde{\mathbf{t}}; \beta_k) + L\langle \mathbf{t}_{k+1} - \mathbf{t}_k, \tilde{\mathbf{t}}\rangle \tag{2.25}$$

Letting $\alpha = \mathbf{t}_{k+1}$ and $\beta = \beta_k$ in (2.9), we get

$$\mathbf{F}(\mathbf{t}_{k+1}) \leq l_{\mathbf{F}}(\mathbf{t}_{k+1}; \beta_k) + \frac{L}{2}\|\mathbf{t}_{k+1} - \beta_k\|_2^2 \tag{2.26}$$

Applying (2.25) to (2.26), we have

$$\begin{aligned}\mathbf{F}(\mathbf{t}_{k+1}) &\leq l_{\mathbf{F}}(\tilde{\mathbf{t}}; \beta_k) + L\langle \mathbf{t}_{k+1} - \beta_k, \tilde{\mathbf{t}} - \mathbf{t}_{k+1}\rangle + \frac{L}{2}\|\mathbf{t}_{k+1} - \beta_k\|_2^2 \\ &= l_{\mathbf{F}}(\tilde{\mathbf{t}}; \beta_k) + \frac{L}{2}\|\tilde{\mathbf{t}} - \beta_k\|_2^2 - \frac{L}{2}\|\tilde{\mathbf{t}} - \mathbf{t}_{k+1}\|_2^2\end{aligned}$$

Substituting $\tilde{\mathbf{t}} = \frac{\mathbf{t}^* + (s_k - 1)\mathbf{t}_k}{s_k}$ into the above inequality and denoting $\gamma_k := (s_{k-1} - 1)\mathbf{t}_{k-1} - s_{k-1}\mathbf{t}_k$, we obtain

$$\begin{aligned}\mathbf{F}(\mathbf{t}_{k+1}) &\leq \frac{s_k - 1}{s_k} l_{\mathbf{F}}(\mathbf{t}_k; \beta_k) + \frac{1}{s_k} l_{\mathbf{F}}(\mathbf{t}^*; \beta_k) \\ &\quad + \frac{L}{2s_k^2}\|(s_k - 1)\mathbf{t}_k + \mathbf{t}^* - s_k\beta_k\|_2^2 - \frac{L}{2s_k^2}\|(s_k - 1)\mathbf{t}_k + \mathbf{t}^* - s_k\mathbf{t}_{k+1}\|_2^2 \\ &= \frac{s_k - 1}{s_k} l_{\mathbf{F}}(\mathbf{t}_k; \beta_k) + \frac{1}{s_k} l_{\mathbf{F}}(\mathbf{t}^*; \beta_k) + \frac{L}{2s_k^2}\|\mathbf{t}^* - \gamma_k\|_2^2 - \frac{L}{2s_k^2}\|\mathbf{t}^* - \gamma_{k+1}\|_2^2 \\ &\leq \frac{s_k - 1}{s_k}\mathbf{F}(\mathbf{t}_k) + \frac{1}{s_k}\mathbf{F}(\mathbf{t}^*) + \frac{L}{2s_k^2}\|\mathbf{t}^* - \gamma_k\|_2^2 - \frac{L}{2s_k^2}\|\mathbf{t}^* - \gamma_{k+1}\|_2^2\end{aligned} \tag{2.27}$$

Here, the first inequality follows from that $l_{\mathbf{F}}$ is convex and $s_k \geq 1$ which is easy to verify. The last inequality follows from the definition of $l_{\mathbf{F}}$ and the convexity of $\mathbf{F}_2$.

Subtracting $\mathbf{F}(\mathbf{t}^*)$ from both sides of the last inequality of (2.27), multiplying s_k^2 at both sides and noticing that $s_{k-1}^2 = s_k(s_k - 1)$, we have

$$s_k^2(\mathbf{F}(\mathbf{t}_{k+1}) - \mathbf{F}(\mathbf{t}^*)) \leq s_{k-1}^2(\mathbf{F}(\mathbf{t}_k)) - \mathbf{F}(\mathbf{t}^*)) + \frac{L}{2}\|\mathbf{t}^* - \gamma_k\|_2^2 - \frac{L}{2}\|\mathbf{t}^* - \gamma_{k+1}\|_2^2$$

Telescoping on the above inequality, and using $s_{-1} = 0$,$\gamma_0 = \mathbf{t}_0$, we have

$$s_k^2(\mathbf{F}(\mathbf{t}_{k+1}) - \mathbf{F}(\mathbf{t}^*)) \leq \frac{L}{2}\|\mathbf{t}^* - \mathbf{t}_0\|_2^2$$

Hence we have (2.21) and (2.22) follows from the triangle inequality.

The conclusion that $\mathbf{t}_k \to \mathbf{t}^*$ when $\mathbf{t}^*$ is the unique minimizer of $\mathbf{F}$ follows directly from Proposition 2.1.1. ■

As we can see from Theorem 2.2.1 the accelerated proximal gradient (APG) algorithm (see also the FISTA algorithm of [1]) is much more efficient than the proximal forward-backward splitting algorithm because it only requires $O(L/\sqrt{\epsilon})$ iterations to obtain an ϵ-optimal solution.

Chapter 3

Exact Recovery

3.1 Analysis

In this Chapter, we will focus on the special case where the given data are $\mathbf{P}_\Lambda \mathbf{f}$ in the image domain and $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in the transform domain, i.e., we have $\mathbf{x} = \mathbf{P}_\Lambda \mathbf{f}$ and $\mathbf{y} = \mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in (1.1). A sufficient condition is given to assert the exact recovery of the underlying data $\mathbf{f}$. We will also propose an algorithm to recover the data and compare it with the algorithm (1.5) proposed in [5].

For simplicity, we denote that $\Gamma^c = \mathcal{M}\backslash\Gamma$ and $\Lambda^c = \mathcal{N}\backslash\Lambda$. The main result of this Chapter is the following Theorem:

Theorem 3.1.1. *Let* $\mathbf{W}$ *be the tight frame transform, i.e.* $\mathbf{W}^T\mathbf{W} = \mathbf{I}$. *Given* $\mathbf{P}_\Lambda \mathbf{f}$ *and* $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ *for some* $\mathbf{f} \in \mathbb{R}^n$, *if the the following inequality*

$$\sum_{i\in\Gamma^c}\sum_{j\in\Lambda^c} |\mathbf{W}(i,j)|^2 < 1 \tag{3.1}$$

holds, then

$$\mathbf{f} = (\mathbf{I} - \mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c})^{-1}(\mathbf{W}^T\mathbf{P}_\Gamma\mathbf{W}\mathbf{f} + \mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_\Lambda\mathbf{f}) \tag{3.2}$$

or

$$\mathbf{f} = (\mathbf{I} - \mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W})^{-1}(\mathbf{P}_\Lambda\mathbf{f} + \mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}) \tag{3.3}$$

The key issue to prove the above Theorem is to prove the invertibility of the operator $\mathbf{I} - \mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}$ in (3.2) and $\mathbf{I} - \mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}$ in (3.3).

It is a common sense that we cannot lose too much information if we want to recover the original data. Therefore the condition (3.1) should guarantee that large proportion of data in the image or transform domain are known. This fact is revealed by the following proposition which is some kind of uncertainty principle with the same spirit of Theorem 2 in [19].

Proposition 3.1.1. *Let* $\mathbf{W}$ *is a tight frame transform,* $\Lambda \subset \mathcal{N}$ *and* $\Gamma \subset \mathcal{M}$ *be given sets. If there exist none zero* $\mathbf{f} \in \mathbb{R}^n$*,* $\varepsilon, \eta \geq 0$*, such that*

$$\|\mathbf{P}_\Lambda\mathbf{f}\|_2 \leq \varepsilon\|\mathbf{f}\|_2 \tag{3.4}$$

and

$$\|\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}\|_2 \leq \eta\|\mathbf{W}\mathbf{f}\|_2 \tag{3.5}$$

hold, then we have

$$\sum_{i\in\Gamma^c}\sum_{j\in\Lambda^c}|\mathbf{W}(i,j)|^2 \geq (1-\varepsilon-\eta)^2$$

Proof. The proof is straitforward, we observed that

$$\|\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f}\|_2^2 = \sum_{i\in\Gamma^c}|(\mathbf{W}\mathbf{P}_{\Gamma^c}\mathbf{f})(i)|^2 = \sum_{i\in\Gamma^c}|\sum_{j=1}^{N}\mathbf{W}(i,j)(\mathbf{P}_{\Lambda^c}\mathbf{f})(j)|^2 = \sum_{i\in\Gamma^c}|\sum_{j\in\Lambda^c}\mathbf{W}(i,j)\mathbf{f}(j)|^2$$

By the Cauchy-Schwarz inequality, for every $i \in \Gamma^c$, we have

$$|\sum_{j\in\Lambda^c}\mathbf{W}(i,j)\mathbf{f}(j)|^2 \leq (\sum_{j\in\Lambda^c}|\mathbf{W}(i,j)|^2)(\sum_{j\in\Lambda^c}|\mathbf{f}(j)|^2) \leq (\sum_{j\in\Lambda^c}|\mathbf{W}(i,j)|^2)\|\mathbf{f}\|_2^2$$

Hence we have

$$\|\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f}\|_2 \leq (\sum_{i\in\Gamma^c}\sum_{j\in\Lambda^c}|\mathbf{W}(i,j)|^2)^{1/2}\|\mathbf{f}\|_2 \tag{3.6}$$

On the other hand, by triangle inequality, we have

$$\begin{aligned}
\|\mathbf{W}\mathbf{f}\|_2 - \|\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f}\|_2 &\leq \|\mathbf{W}\mathbf{f} - \mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f}\|_2 \\
&\leq \|\mathbf{W}\mathbf{f} - \mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{f}\|_2 + \|\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{f} - \mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f}\|_2 \\
&\leq \eta\|\mathbf{f}\|_2 + \|\mathbf{P}_{\Gamma^c}\|\|\mathbf{W}\|\|\mathbf{f} - \mathbf{P}_{\Lambda^c}\mathbf{f}\|_2 \\
&\leq \eta\|\mathbf{f}\|_2 + \varepsilon\|\mathbf{f}\|_2
\end{aligned}$$

where the operator norm $\|\mathbf{P}_{\Gamma^c}\|, \|\mathbf{W}\|$ are both equal to 1. The last two inequality come from the conditions(3.4),(3.5) respectively.

Combine this with (3.6), and use the property $\|\mathbf{W}\mathbf{f}\|_2 = \|\mathbf{f}\|_2$, we can get that

$$\begin{aligned}
(\sum_{i\in\Gamma^c}\sum_{j\in\Lambda^c}|\mathbf{W}(i,j)|^2)^{1/2}\|\mathbf{f}\|_2 &\geq \|\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f}\|_2 \\
&\geq \|\mathbf{W}\mathbf{f}\|_2 - \eta\|\mathbf{f}\|_2 - \varepsilon\|\mathbf{f}\|_2 \\
&= (1-\eta-\varepsilon)\|\mathbf{f}\|_2
\end{aligned}$$

By dividing $\|\mathbf{f}\|_2$ on both sides, we get the result. ■

Proposition 3.1.1 shows that if (3.4) and (3.5) hold with small ϵ and η, i.e. there is too little information given, the chance for (3.1) to be hold is slim, i.e. the exact recovery via Theorem 3.1.1 becomes difficult .

In the following we prove Theorem 3.1.1.

Proof of Thm. 3.1.1. First of all, from (3.1) and (3.6), we have the operator norm $\|\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\| < 1$, which ensures the invertibility of $\mathbf{I} - \mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}$.

Furthermore, we observe that

$$\begin{aligned}(\mathbf{I}-\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c})\mathbf{f} &= \mathbf{W}^T(\mathbf{I}-\mathbf{P}_{\Gamma^c})\mathbf{W}\mathbf{f}+\mathbf{W}^T(\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{f}-\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}\mathbf{f})\\ &= \mathbf{W}^T\mathbf{P}_{\Gamma}\mathbf{W}\mathbf{f}+\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}(\mathbf{I}-\mathbf{P}_{\Lambda^c})\mathbf{f}\\ &= \mathbf{W}^T\mathbf{P}_{\Gamma}\mathbf{W}\mathbf{f}+\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda}\mathbf{f}\end{aligned}$$

We notice that the data at the right hand side of the above equality are known since the data $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ and $\mathbf{P}_\Lambda\mathbf{f}$ are both given. Then by the invertibility of $\mathbf{I}-\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}$, we can restore the original $\mathbf{f}$ from the known data $\mathbf{P}_\Lambda\mathbf{f}$ and $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ by the following expression

$$\mathbf{f}=(\mathbf{I}-\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c})^{-1}(\mathbf{W}^T\mathbf{P}_{\Gamma}\mathbf{W}\mathbf{f}+\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda}\mathbf{f}) \tag{3.7}$$

The above equality provides us the algorithm for exactly recovering the original signal $\mathbf{f}$. Furthermore, there is another expression which also provides a way to recover $\mathbf{f}$.

Similarly, we have the operator norm $\|\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\|<1$, which ensures the invertibility of $\mathbf{I}-\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}$. By a similar argument, we observe that

$$\begin{aligned}(\mathbf{I}-\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W})\mathbf{f} &= (\mathbf{I}-\mathbf{P}_{\Lambda^c})\mathbf{f}+\mathbf{P}_{\Lambda^c}(\mathbf{f}-\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{f})\\ &= \mathbf{P}_{\Lambda}\mathbf{f}+\mathbf{P}_{\Lambda^c}\mathbf{W}^T(\mathbf{I}-\mathbf{P}_{\Gamma^c})\mathbf{W}\mathbf{f}\\ &= \mathbf{P}_{\Lambda}\mathbf{f}+\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma}\mathbf{W}\mathbf{f}\end{aligned}$$

The data at the right hand side of the above equality are known since $\mathbf{P}_\Lambda\mathbf{f}$ and $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ are both given. Then by the invertibility of $\mathbf{I}-\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}$, we can restore the original $\mathbf{f}$ from the known data $\mathbf{P}_\Lambda\mathbf{f}$ and $\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}$ by the expression

$$\mathbf{f}=(\mathbf{I}-\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W})^{-1}(\mathbf{P}_{\Lambda}\mathbf{f}+\mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma}\mathbf{W}\mathbf{f}) \tag{3.8}$$

Thus, we proved the theorem and provided two ways to recover the original

data. ■

The exact recovery by applying Theorem 3.1.1 necessarily implies that the recovery is unique. Indeed, this uniqueness is asserted by the assumption (3.1). Assume that $\mathbf{f}_1$ and $\mathbf{f}_2$ both satisfy (1.1), we let $\mathbf{h} = \mathbf{f}_1 - \mathbf{f}_1$. Then, it follows from (1.1) that $\mathbf{P}_\Lambda \mathbf{h} = \mathbf{P}_\Gamma \mathbf{W}\mathbf{h} = 0$, which implies by (3.2) or (3.3) $\mathbf{h} = 0$ and hence $\mathbf{f}_1 = \mathbf{f}_2$. This uniqueness is why we called exact recovery in this thesis.

3.2 Algorithms

Theorem 3.1.1 guarantees the exact recovery of $\mathbf{f}$ under the condition (3.1), and (3.2) and (3.3) provide the recovery expressions of $\mathbf{f}$. However we merely use them to reconstruct the original data in the implementations since it is complicated to compute the inverse of a matrix. In the following, we will propose the iterative algorithms for the two recovery expressions above.

The algorithms are based on the following well known approach (see for instance [24]). Let $\mathbf{L}$ be a $m \times m$ matrix that satisfies $\|\mathbf{L}\| < 1$. Then the matrix $\mathbf{I} - \mathbf{L}$ is invertible. Starting from any given vector $\mathbf{v}_0 \in \mathbb{R}^m$, we define

$$\mathbf{v}_{k+1} = \mathbf{v}_0 + \mathbf{L}\mathbf{v}_k, \quad k \geq 0.$$

The contraction mapping principle ensures that this iterative procedure converges in a geometric rate to the unique fixed point $\mathbf{v}^*$ given by

$$\mathbf{v}^* = \mathbf{v}_0 + \mathbf{L}\mathbf{v}^*.$$

In other words, $\mathbf{v}^* = (\mathbf{I} - \mathbf{L})^{-1}\mathbf{v}_0$.

Thus, for (3.2), by letting $\mathbf{L} = \mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}\mathbf{P}_{\Lambda^c}$ and $\mathbf{v}_0 = \mathbf{W}^T\mathbf{P}_\Gamma\mathbf{W}\mathbf{f} + \mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{P}_\Lambda\mathbf{f}$,

we will finally get the iterative algorithm :

$$\mathbf{f}_{k+1} = \mathbf{W}^T(\mathbf{P}_\Gamma \mathbf{W}\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}(\mathbf{P}_\Lambda \mathbf{f} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{f}_k)) \tag{3.9}$$

In (3.9), the data is updated in the following procedure: in the image domain, we update the coefficients on Λ by the known data $\mathbf{P}_\Lambda \mathbf{f}$, then transform it to the transform domain and use the known data $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in the transform domain to replace the coefficients on Γ. Then we transform it back to the image domain and go to the next iteration.

In the same way, we can formulate the iterative algorithm for (3.3) by letting $\mathbf{L} = \mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_{\Gamma^c}\mathbf{W}$ and $\mathbf{v}_0 = \mathbf{P}_\Lambda \mathbf{f} + \mathbf{P}_{\Lambda^c}\mathbf{W}^T\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$:

$$\mathbf{f}_{k+1} = \mathbf{P}_\Lambda \mathbf{f} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{W}^T(\mathbf{P}_\Gamma \mathbf{W}\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k) \tag{3.10}$$

In (3.10), from $\mathbf{f}_k$ to $\mathbf{f}_{k+1}$, we firstly transform $\mathbf{f}_k$ to the transform domain to get the transform coefficients $\mathbf{W}\mathbf{f}_k$. Then we replace the coefficients on Γ by the known data $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$. After that, we transform it back to the image domain and update the data on Λ by the known data $\mathbf{P}_\Lambda \mathbf{f}$ in the image domain.

We have provided two algorithms (3.9) and (3.10) for the exact recovery case. However, we will see that they are essentially the same by the following arguments.

In (3.9), if we denote $\mathbf{t}_k = \mathbf{x} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{f}_k$, then our algorithm can be written as follows

$$\begin{cases} \mathbf{t}_k = \mathbf{x} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{f}_k \\ \mathbf{r}_k = \mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{t}_k \\ \mathbf{f}_{k+1} = \mathbf{W}^T\mathbf{r}_n \end{cases} \tag{3.11}$$

Similarly, in (3.10), we denote $\mathbf{r}_k = \mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k$. The algorithm (2.15) is

equivalent to the following iteration

$$\begin{cases} \mathbf{r}_k = \mathbf{y} + (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{W}\mathbf{f}_k \\ \mathbf{t}_k = \mathbf{W}^T\mathbf{r}_k \\ \mathbf{f}_{k+1} = \mathbf{x} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{t}_k \end{cases} \tag{3.12}$$

By comparing (3.11) and (3.12), we can easily see that these two iterations are essentially doing the same calculation process. Thus the algorithm (3.9) and (3.10) can be considered as the same. The only difference is that we choose $\mathbf{W}^T\mathbf{r}$ as the result in (3.9) and $\mathbf{P}_\Lambda\mathbf{f} + (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{t}$ in (3.10) where $\mathbf{r}$ and $\mathbf{t}$ are the limit of $\mathbf{r}_k$ in(3.11) and $\mathbf{t}_k$ in (3.12) respectively. Therefore, we can just consider (3.10) as our exact recovery algorithm.

Note that the exact recovery algorithm (3.10) is slightly different from the balanced approach algorithm (1.5) by a plunging denoising operator, i.e., soft thresholding $\mathbf{T}_u$. This is reasonable since in practise the given data are mostly contaminated by noises.

Chapter 4

Analysis Based Approach

4.1 Analysis and Algorithm

In the general case, the data $\mathbf{x}$ in the image domain and $\mathbf{y}$ in the transform domain are arbitrarily given and are normally contaminated by noises. Furthermore, the sufficient condition (3.1) that guarantees the exact recovery does not hold with high probability. In that case, the two domains image restoration problem

$$\begin{cases} \mathbf{P}_\Lambda \mathbf{f} = \mathbf{x} \\ \mathbf{P}_\Gamma \mathbf{W} \mathbf{f} = \mathbf{y} \end{cases}$$

is a very ill posed problem in general. It may have no solution or have infinity many solutions in many cases and since the data is normally contaminated by noises, it is unnecessary to have exact recovery in those cases.

We propose the following analysis based model under the assumptions that the data given is somehow close to $\mathbf{P}_\Lambda \mathbf{f}$ in the image domain and $\mathbf{P}_\Gamma \mathbf{W} \mathbf{f}$ in the transform domain where $\mathbf{f}$ is the underlying solution which has a sparse approximation by the canonical coefficients $\mathbf{W}\mathbf{f}$. The solution is a minimizer of the following

minimization problem:

$$\min_{\mathbf{f}\in\mathbb{R}^n}\{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{W}\mathbf{f}\|_1\} \tag{4.1}$$

where $\nu > 0$ is a weighted parameter and μ is a positively weighted vector.

Now we consider each terms of (4.1): The first term penalizes the distant between $\mathbf{f}$ and the given data in the image domain. The second term penalizes the distant of our solution to the known data in the transform domain. Thus the first two terms guarantee the fidelity of the solution. The last term uses the assumptions that the underlying solution has a good sparse approximation by its canonical coefficients.

Since the minimization problem (4.1) is not separable, it cannot be solved simply by thresholding as in the balanced or synthesis based approach (see, e.g., [7, 11, 21, 22]). In the next we will derive an iterative algorithm for (4.1) by using split Bregman method.

The derivation of the split Bregman method in [8, 25] is based on Bregman distance. Furthermore, the split Bregman method can be understood as the augmented Lagrangian method (see [23]) applying to (4.1)(see e.g.,[20, 33]). It is clear that (4.1) is equivalent to the following minimization problem

$$\begin{aligned}&\min_{\mathbf{f}\in\mathbb{R}^n,\mathbf{d}\in\mathbb{R}^m}\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{d}-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{d}\|_1+\frac{\lambda}{2}\|\mathbf{d}-\mathbf{W}\mathbf{f}\|_2^2\\ &\text{subject to } \mathbf{W}\mathbf{f}=\mathbf{d}\end{aligned} \tag{4.2}$$

where $\lambda > 0$. The Lagrangian for problem (4.2) is given by

$$\mathcal{L}(\mathbf{f},\mathbf{d},\mathbf{p})=\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{d}-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{d}\|_1+\frac{\lambda}{2}\|\mathbf{d}-\mathbf{W}\mathbf{f}\|_2^2+\langle\mathbf{p},\mathbf{d}-\mathbf{W}\mathbf{f}\rangle$$

The saddle points of $\mathcal{L}(\mathbf{f},\mathbf{d},\mathbf{p})$ are obtained by the following iteration

$$\begin{cases} (\mathbf{f}_{k+1},\mathbf{d}_{k+1}) = \arg\min_{\mathbf{f},\mathbf{d}}\{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{d}-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{d}\|_1 \\ \qquad\qquad +\frac{\lambda}{2}\|\mathbf{d}-\mathbf{W}\mathbf{f}\|_2^2+\langle\mathbf{p}_k,\mathbf{d}-\mathbf{W}\mathbf{f}\rangle\} \\ \mathbf{p}_{k+1} = \mathbf{p}_k+\lambda(\mathbf{d}_{k+1}-\mathbf{W}\mathbf{f}_{k+1}) \end{cases}$$

By letting $\mathbf{b}_k = -\mathbf{p}_k/\lambda$, the above iteration becomes

$$\begin{cases} (\mathbf{f}_{k+1},\mathbf{d}_{k+1}) = \arg\min_{\mathbf{f},\mathbf{d}}\{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{d}-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{d}\|_1 \\ \qquad\qquad +\frac{\lambda}{2}\|\mathbf{W}\mathbf{f}-\mathbf{d}+\mathbf{b}_k\|_2^2\} \\ \mathbf{b}_{k+1} = \mathbf{b}_k+(\mathbf{W}\mathbf{f}_{k+1}-\mathbf{d}_{k+1}) \end{cases}$$

For the first subproblem above, we usually use alternative minimization method to solve it. Then, we can get the following iteration for (4.1)

$$\begin{cases} \mathbf{f}_{k+1} = \arg\min_{\mathbf{f}}\{\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}-\mathbf{x}\|_2^2+\frac{\lambda}{2}\|\mathbf{W}\mathbf{f}-\mathbf{d}_k+\mathbf{b}_k\|_2^2\} \\ \mathbf{d}_{k+1} = \arg\min_{\mathbf{d}}\{\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{d}-\mathbf{y}\|_2^2+\frac{\lambda}{2}\|\mathbf{W}\mathbf{f}_{k+1}-\mathbf{d}+\mathbf{b}_k\|_2^2+\|\mathrm{diag}(\mu)\mathbf{d}\|_1\} \\ \mathbf{b}_{k+1} = \mathbf{b}_k+(\mathbf{W}\mathbf{f}_{k+1}-\mathbf{d}_{k+1}) \end{cases} \tag{4.3}$$

The first subproblem is easy to solve and implement. For the second subproblem, since it is separable for $\mathbf{d}$, we can handle it for two parts, one is $\{\mathbf{d}(i)\}_{i\notin\Gamma}$ and the other one $\{\mathbf{d}(i)\}_{i\in\Gamma}$ where $\mathbf{d}(i)$ is the i-th element of the vector $\mathbf{d}$. We will show that for both cases, the solution of the subproblem is simply a soft thresholding. Now let us handle them separately.

For those $i\notin\Gamma$, the second subproblem of (4.3) simply becomes

$$\mathbf{d}_{k+1}(i) = \arg\min_{\mathbf{d}(i)\in\mathbb{R}}\{\frac{\lambda}{2}(\mathbf{W}\mathbf{f}_{k+1}(i)-\mathbf{d}(i)+\mathbf{b}_k(i))^2+|\mu(i)\mathbf{d}(i)|\}$$

and it is well-known that the solution is $\mathbf{d}_{k+1}(i) = t_{\frac{\mu(i)}{\lambda}}(\mathbf{W}\mathbf{f}_{k+1}(i)+\mathbf{b}_k(i))$ where

$t_{\frac{\mu(i)}{\lambda}}$ is the i-th component of the theresholding operator $\mathbf{T}_{\frac{\mu}{\lambda}}$ (see, e.g. [13]).

For those $i \in \Gamma$, the second subproblem of (4.3) is

$$\mathbf{d}_{k+1}(i) = \arg\min_{\mathbf{d}(i)\in\mathbb{R}}\{\frac{\nu}{2}(\mathbf{d}(i) - \mathbf{y}(i))^2 + \frac{\lambda}{2}(\mathbf{W}\mathbf{f}_{k+1}(i) - \mathbf{d}(i) + \mathbf{b}_k(i))^2 + |\mu(i)\mathbf{d}(i)|\}$$

which is also a soft thresholding by the following argument:

$$\begin{aligned}\mathbf{d}_{k+1}(i) &= \arg\min_{\mathbf{d}(i)\in\mathbb{R}}\{\frac{\nu}{2}(\mathbf{d}(i) - \mathbf{y}(i))^2 + \frac{\lambda}{2}(\mathbf{W}\mathbf{f}_{k+1}(i) - \mathbf{d}(i) + \mathbf{b}_k(i))^2 + |\mu(i)\mathbf{d}(i)|\} \\ &= \arg\min_{\mathbf{d}(i)\in\mathbb{R}}\{|\mu(i)\mathbf{d}(i)| + \frac{\nu+\lambda}{2}[\mathbf{d}(i) - \frac{\nu\mathbf{y}(i) + \lambda(\mathbf{W}\mathbf{f}_{k+1}(i) + \mathbf{b}_k(i))}{\nu+\lambda}]^2\} \\ &= t_{\frac{\mu(i)}{\nu+\lambda}}(\frac{\nu\mathbf{y}(i) + \lambda(\mathbf{W}\mathbf{f}_{k+1}(i) + \mathbf{b}_k(i))}{\nu+\lambda})\end{aligned}$$

Combine these two cases, we can write the solution for the second subproblem of (4.3) in the following form

$$\mathbf{d}_{k+1} = (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{T}_{\frac{\mu}{\lambda}}(\mathbf{W}\mathbf{f}_{k+1} + \mathbf{b}_k) + \mathbf{P}_\Gamma\mathbf{T}_{\frac{\mu}{\nu+\lambda}}(\frac{\nu\mathbf{y} + \lambda(\mathbf{W}\mathbf{f}_{k+1} + \mathbf{b}_k)}{\nu+\lambda})$$

Hence, algorithm (4.3) can be written by a more explicit way as follows:

$$\begin{cases}\mathbf{f}_{k+1} = (\mathbf{P}_\Lambda + \lambda\mathbf{I})^{-1}(\mathbf{x} + \lambda\mathbf{W}^T(\mathbf{d}_k - \mathbf{b}_k)) \\ \mathbf{d}_{k+1} = (\mathbf{I} - \mathbf{P}_\Gamma)\mathbf{T}_{\frac{\mu}{\lambda}}(\mathbf{W}\mathbf{f}_{k+1} + \mathbf{b}_k) + \mathbf{P}_\Gamma\mathbf{T}_{\frac{\mu}{\nu+\lambda}}(\frac{\nu\mathbf{y}+\lambda(\mathbf{W}\mathbf{f}_{k+1}+\mathbf{b}_k)}{\nu+\lambda}) \\ \mathbf{b}_{k+1} = \mathbf{b}_k + (\mathbf{W}\mathbf{f}_{k+1} - \mathbf{d}_{k+1})\end{cases} \tag{4.4}$$

where $\mathbf{T}_\mu$ is the soft threshloding operator as defined before.

Since $\mathbf{P}_\Lambda + \lambda\mathbf{I}$ is a diagonal matrix, it is easy to be inverted and computed. Thus the algorithm (4.4) is easy to be implemented and efficient.

For the algorithm (4.3), we have the following convergence result which is similar to Theorem 4.8 in [16]: Assume there is at least one solution $\mathbf{f}^*$ for (4.1),

then we have the following property for the split Bregman algorithm (4.3)

$$\begin{aligned}\lim_{k\to\infty}&\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}_k-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}_k-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{W}\mathbf{f}_k\|_1\\ =&\frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f}^*-\mathbf{x}\|_2^2+\frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{W}\mathbf{f}^*-\mathbf{y}\|_2^2+\|\mathrm{diag}(\mu)\mathbf{W}\mathbf{f}^*\|_1\end{aligned}$$

Furthermore, when (4.1) has a unique solution, we have $\lim_{k\to\infty}\|\mathbf{f}_k-\mathbf{f}^*\|_2=0$.

The result will be proved in the next section in a more general context.

4.2 Convergence Analysis

In [16], the authors give the convergence analysis of the split Bregman method for the problem

$$\min_{\mathbf{f}\in\mathbb{R}^n}\{H(\mathbf{f})+\|\mathrm{diag}(\mu)\mathbf{W}\mathbf{f}\|_1\}$$

where $H(\mathbf{f})$ is convex and smooth. The convergence analysis of (4.3) is quite similar. For consistence of this thesis, I will prove the convergence of algorithm (4.3).

To get the convergence result for the iteration (4.3), we will provide a more general one. We consider the following minimization problem

$$\min_{\mathbf{f}\in\mathbb{R}^n}\{H(\mathbf{f})+K(\mathbf{W}\mathbf{f})\} \tag{4.5}$$

where $H(\mathbf{f})$ is a smooth convex function and $K(\mathbf{d})$ is convex. By a similar argument, we will get the split Bregman algorithm for (4.5)

$$\begin{cases}\mathbf{f}_{k+1}=\arg\min_{\mathbf{f}}\{H(\mathbf{f})+\frac{\lambda}{2}\|\mathbf{W}\mathbf{f}-\mathbf{d}_k+\mathbf{b}_k\|_2^2\}\\ \mathbf{d}_{k+1}=\arg\min_{\mathbf{d}}\{K(\mathbf{d})+\frac{\lambda}{2}\|\mathbf{W}\mathbf{f}_{k+1}-\mathbf{d}+\mathbf{b}_k\|_2^2\}\\ \mathbf{b}_{k+1}=\mathbf{b}_k+(\mathbf{W}\mathbf{f}_{k+1}-\mathbf{d}_{k+1})\end{cases} \tag{4.6}$$

We have the following theorem which is a generalization of Theorem 4.8 in [16]

Theorem 4.2.1. *For the iteration algorithm* (4.6) *generated by split Bregman method, if there is at least one solution* $\mathbf{f}^*$ *for the minimization problem* (4.5) *and* $\lambda > 0$*, we have*

$$\lim_{k\to\infty}\{H(\mathbf{f}_k) + K(\mathbf{W}\mathbf{f}_k)\} = \{H(\mathbf{f}^*) + K(\mathbf{W}\mathbf{f}^*)\} \tag{4.7}$$

and if the solution is unique, then we have

$$\lim_{k\to\infty} \|\mathbf{f}_k - \mathbf{f}^*\|_2^2 = 0 \tag{4.8}$$

By substituting $H(\mathbf{f}) = \frac{1}{2}\|\mathbf{P}_\Lambda\mathbf{f} - \mathbf{x}\|_2^2$ and $K(\mathbf{d}) = \frac{\nu}{2}\|\mathbf{P}_\Gamma\mathbf{d} - \mathbf{y}\|_2^2 + \|\mathrm{diag}(\mu)\mathbf{d}\|_1$ in theorem 4.2.1, we can easily get the convergence result for the analysis based algorithm (4.3). We now give a proof of the theorem

Proof. The first order optimality condition of (4.6) gives

$$\begin{cases} 0 = \nabla H(\mathbf{f}_{k+1}) + \lambda\mathbf{W}^T(\mathbf{W}\mathbf{f}_{k+1} - \mathbf{d}_k + \mathbf{b}_k) \\ 0 = \mathbf{p}_{k+1} + \lambda(\mathbf{d}_{k+1} - \mathbf{W}\mathbf{f}_{k+1} - \mathbf{b}_k) \quad \text{with} \quad \mathbf{p}_{k+1} \in \partial K(\mathbf{d}_{k+1}) \\ \mathbf{b}_{k+1} = \mathbf{b}_k + (\mathbf{W}\mathbf{f}_{k+1} - \mathbf{d}_{k+1}) \end{cases} \tag{4.9}$$

where $\partial K(\mathbf{d}_{k+1})$ is the subdifferential of $K(\mathbf{d})$ at $\mathbf{d}_{k+1}$.

The solution $\mathbf{f}^*$ for (4.5) must satisfies

$$0 = \mathbf{W}^T\mathbf{p}^* + \nabla H(\mathbf{f}^*) \tag{4.10}$$

where $\mathbf{p}^* \in \partial K(\mathbf{d}^*)$ with $\mathbf{d}^* = \mathbf{W}\mathbf{f}^*$. Letting $\mathbf{b}^* = \frac{1}{\lambda}\mathbf{p}^*$, we have

$$\begin{cases} 0 = \nabla H(\mathbf{f}^*) + \lambda\mathbf{W}^T(\mathbf{W}\mathbf{f}^* - \mathbf{d}^* + \mathbf{b}^*) \\ 0 = \mathbf{p}^* + \lambda(\mathbf{d}^* - \mathbf{W}\mathbf{f}^* - \mathbf{b}^*) \quad \text{with} \quad \mathbf{p}^* \in \partial K(\mathbf{d}_{k+1}) \\ \mathbf{b}^* = \mathbf{b}^* + (\mathbf{W}\mathbf{f}^* - \mathbf{d}^*) \end{cases} \tag{4.11}$$

This means that $(\mathbf{f}^*, \mathbf{d}^*, \mathbf{b}^*)$ is a fixed point of (4.9). Denote the errors by

$$\mathbf{f}_k^e = \mathbf{f}_k - \mathbf{f}^*, \quad \mathbf{d}_k^e = \mathbf{d}_k - \mathbf{d}^*, \quad \mathbf{b}_k^e = \mathbf{b}_k - \mathbf{b}^*$$

Subtracting the first equation of (4.11) from the first equation of (4.9), we obtain

$$0 = \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*) + \lambda \mathbf{W}^T(\mathbf{W}\mathbf{f}_{k+1}^e - \mathbf{d}_k^e + \mathbf{b}_k^e)$$

Take inner product on both side with respect to $\mathbf{f}_{k+1}^e$, we have

$$0 = \langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}_{k+1}^e \rangle + \lambda \|\mathbf{W}\mathbf{f}_{k+1}^e\|_2^2 - \lambda \langle \mathbf{W}^T\mathbf{d}_k^e, \mathbf{f}_{k+1}^e \rangle + \lambda \langle \mathbf{W}^T\mathbf{b}_k^e, \mathbf{f}_{k+1}^e \rangle \tag{4.12}$$

Applying similar manipulations to the second equation of (4.11) and (4.9), we have

$$0 = \langle \mathbf{p}_{k+1} - \mathbf{p}^*, \mathbf{d}_{k+1}^e \rangle + \lambda \|\mathbf{d}_{k+1}^e\|_2^2 - \lambda \langle \mathbf{W}\mathbf{f}_{k+1}^e, \mathbf{d}_{k+1}^e \rangle - \lambda \langle \mathbf{b}_k^e, \mathbf{d}_{k+1}^e \rangle \tag{4.13}$$

where $\mathbf{p}_{k+1} \in \partial K(\mathbf{d}_{k+1})$ and $\mathbf{p}^* = \lambda \mathbf{b}^* \in \partial K(\mathbf{d}^*)$.

By subtracting the third equation of (4.11) from the third equation of (4.9), we get

$$\mathbf{b}_{k+1}^e = \mathbf{b}_k^e + (\mathbf{W}\mathbf{f}_{k+1}^e - \mathbf{d}_{k+1}^e) \tag{4.14}$$

Taking the inner product of (4.14) with itself, we will have the following identity

$$\langle \mathbf{b}_k^e, \mathbf{W}\mathbf{f}_{k+1}^e - \mathbf{d}_{k+1}^e \rangle = \frac{1}{2}(\|\mathbf{b}_{k+1}^e\|_2^2 - \|\mathbf{b}_k^e\|_2^2) - \frac{1}{2}\|\mathbf{W}\mathbf{f}_{k+1}^e - \mathbf{d}_{k+1}^e\|_2^2 \tag{4.15}$$

By summing (4.13) and (4.12), we get

$$\begin{aligned}0 =&\langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}^e_{k+1}\rangle + \langle \mathbf{p}_{k+1} - \mathbf{p}^*, \mathbf{d}^e_{k+1}\rangle \\ &+ \lambda(\|\mathbf{W}\mathbf{f}^e_{k+1}\|_2^2 + \|\mathbf{d}^e_{k+1}\|_2^2 - \langle \mathbf{W}\mathbf{f}^e_{k+1}, \mathbf{d}^e_{k+1} + \mathbf{d}^e_k\rangle + \langle \mathbf{b}^e_k, \mathbf{W}\mathbf{f}^e_{k+1} - \mathbf{d}^e_{k+1}\rangle)\end{aligned} \tag{4.16}$$

Substituting (4.15) into (4.16), we have

$$\begin{aligned}&\frac{\lambda}{2}(\|\mathbf{b}^e_k\|_2^2 - \|\mathbf{b}^e_{k+1}\|_2^2) \\ &= \langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}^e_{k+1}\rangle + \langle \mathbf{p}_{k+1} - \mathbf{p}^*, \mathbf{d}^e_{k+1}\rangle \\ &+ \lambda(\|\mathbf{W}\mathbf{f}^e_{k+1}\|_2^2 + \|\mathbf{d}^e_{k+1}\|_2^2 - \langle \mathbf{W}\mathbf{f}^e_{k+1}, \mathbf{d}^e_k + \mathbf{d}^e_{k+1}\rangle - \frac{1}{2}\|\mathbf{W}\mathbf{f}^e_{k+1} - \mathbf{d}^e_{k+1}\|_2^2 \\ &= \langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}^e_{k+1}\rangle + \langle \mathbf{p}_{k+1} - \mathbf{p}^*, \mathbf{d}^e_{k+1}\rangle \\ &+ \lambda(\frac{1}{2}\|\mathbf{W}\mathbf{f}^e_{k+1}\|_2^2 + \frac{1}{2}\|\mathbf{d}^e_{k+1}\|_2^2 - \langle \mathbf{W}\mathbf{f}^e_{k+1}, \mathbf{d}^e_k\rangle) \\ &= \langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}^e_{k+1}\rangle + \langle \mathbf{p}_{k+1} - \mathbf{p}^*, \mathbf{d}^e_{k+1}\rangle \\ &+ \lambda(\frac{1}{2}\|\mathbf{W}\mathbf{f}^e_{k+1} - \mathbf{d}^e_k\|_2^2 + \frac{1}{2}\|\mathbf{d}^e_{k+1}\|_2^2 - \frac{1}{2}\|\mathbf{d}^e_k\|_2^2)\end{aligned} \tag{4.17}$$

By summing the above equation from $k = 0$ to $k = K$, we get

$$\begin{aligned}&\frac{\lambda}{2}(\|\mathbf{b}^e_0\|_2^2 - \|\mathbf{b}^e_{K+1}\|_2^2 + \|\mathbf{d}^e_0\|_2^2) \\ &= \sum_{k=0}^{K}\langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}_{k+1} - \mathbf{f}^*\rangle + \sum_{k=0}^{K}\langle \mathbf{p}_{k+1} - \mathbf{p}^*, \mathbf{d}_{k+1} - \mathbf{d}^*\rangle \\ &+ \lambda(\frac{1}{2}\sum_{k=0}^{K}\|\mathbf{W}\mathbf{f}^e_{k+1} - \mathbf{d}^e_k\|_2^2 + \frac{1}{2}\|\mathbf{d}^e_{K+1}\|_2^2)\end{aligned} \tag{4.18}$$

By the property of subgradient, all the terms at the right-hand side of the above identity are nonnegative. Thus we have the following inequality:

$$\frac{\lambda}{2}(\|\mathbf{b}^e_0\|_2^2 + \|\mathbf{d}^e_0\|_2^2) \geq \sum_{k=0}^{K}\langle \nabla H(\mathbf{f}_{k+1}) - \nabla H(\mathbf{f}^*), \mathbf{f}_{k+1} - \mathbf{f}^*\rangle \tag{4.19}$$

Since $\lambda > 0$, we have $\sum_{k=0}^{K}\langle \nabla H(u_{k+1}) - \nabla H(u^*), u_{k+1} - u^*\rangle < +\infty$, which leads

to

$$\lim_{k\to+\infty}\langle\nabla H(\mathbf{f}_{k+1})-\nabla H(\mathbf{f}^*),\mathbf{f}_{k+1}-\mathbf{f}^*\rangle=0 \tag{4.20}$$

By the definition of subgradient, we have

$$H(\mathbf{f}_k)-H(\mathbf{f}^*)-\langle\mathbf{f}_k-\mathbf{f}^*,\nabla H(\mathbf{f}^*)\rangle\geq 0$$

and

$$H(\mathbf{f}^*)-H(\mathbf{f}_k)-\langle\mathbf{f}^*-\mathbf{f}_k,\nabla H(\mathbf{f}_k)\rangle\geq 0$$

which leads to

$$0\leq H(\mathbf{f}_k)-H(u^*)-\langle\mathbf{f}_k-\mathbf{f}^*,\nabla H(\mathbf{f}^*)\rangle\leq\langle\nabla H(\mathbf{f}_k)-\nabla H(\mathbf{f}^*),\mathbf{f}_k-\mathbf{f}^*\rangle$$

This, together with (4.20), we have

$$\lim_{k\to+\infty}H(\mathbf{f}_k)-H(\mathbf{f}^*)-\langle\mathbf{f}_k-\mathbf{f}^*,\nabla H(\mathbf{f}^*)\rangle=0 \tag{4.21}$$

Similarly, we can have the following results by a similar argument

$$\lim_{k\to+\infty}K(\mathbf{d}_k)-K(\mathbf{d}^*)-\langle\mathbf{d}_k-\mathbf{d}^*,\partial K(\mathbf{d}^*)\rangle=0 \tag{4.22}$$

Furthermore,(4.18) also provides the following inequalities

$$\frac{\mu}{2}(\|\mathbf{b}_0^e\|_2^2+\|\mathbf{d}_0^e\|_2^2)\geq\frac{\mu}{2}\sum_{k=0}^{K}\|\mathbf{W}\mathbf{f}_{k+1}^e-\mathbf{d}_k^e\|_2^2 \tag{4.23}$$

which leads to

$$\lim_{k\to+\infty}\|\mathbf{W}\mathbf{f}_{k+1}^e-\mathbf{d}_k^e\|_2^2=0$$

This, with the fact $\mathbf{W}u^* = \mathbf{d}^*$, we have the following result

$$\lim_{k\to+\infty} \|\mathbf{W}\mathbf{f}_{k+1} - \mathbf{d}_k\|_2^2 = 0 \tag{4.24}$$

Since $\|\cdot\|_1$ and $\|\cdot\|_2$ are both continuous, by (4.22) and (4.24), we have

$$\lim_{k\to+\infty} K(\mathbf{W}\mathbf{f}_k) - K(\mathbf{W}\mathbf{f}^*) - \langle \mathbf{W}\mathbf{f}_k - \mathbf{W}\mathbf{f}^*, p^*)\rangle = 0 \tag{4.25}$$

Summing (4.21) and (4.25), we have

$$\begin{aligned}\lim_{k\to+\infty} (H(\mathbf{f}_k) + K(\mathbf{W}\mathbf{f}_k) - H(\mathbf{f}^*) - K(\mathbf{W}\mathbf{f}^*) \\ - \langle \mathbf{f}_k - \mathbf{f}^*, \nabla H(\mathbf{f}^*) + \mathbf{W}^T p^*\rangle) = 0\end{aligned} \tag{4.26}$$

By (4.10) and (4.26), we proves (4.7) and (4.8) follows from Proposition 2.1.1. ■

Chapter 5

Numerical Implementation

In this Chapter, we will apply our proposed analysis based algorithm on the following three image restoration problems:

(1) Image inpainting

(2) Super resolution image reconstruction with multiple sensors

(3) Super resolution image reconstruction with different zooms

(4) Data reconstruction from normal vectors

The first three applications correspond to three different data missing categories. The second and the fourth application are in the same data category but different problems in practise. The quality of the reconstructed images is evaluated by the peak signal-to-noise ratio(PSNR) defined as

$$\text{PSNR} = 20 \log_{10} \frac{255\sqrt{N}}{\|\mathbf{f} - \mathbf{f}^{\infty}\|_2}$$

where $\mathbf{f}$ and $\mathbf{f}^{\infty}$ are the original image and the reconstructed image respectively, and N is the number of pixels of $\mathbf{f}$. For all our implementations, we set the initial data $\mathbf{f}_0$ be zeros and the iteration is recorded when the reconstructed image achieves the highest PSNR value. Furthermore, in each image restoration problem,

we also illustrate the result by the balanced approach algorithm, APG for the balanced approach and the analysis based approach algorithm.

For the first three application, the transform we used are linear B-spline tight frame, i.e., we use the following filters to construct the transform matrix

$$h_0 = (1/4, 2/4, 1/4),\ h_1 = (-1/4, 2/4, -1/4),\ h_2 = (\sqrt{2}/4, 0, -\sqrt{2}/4)$$

and Haar wavelet in the fourth application, i.e.,

$$h_0 = (1/2, 1/2),\ h_2 = (1/2, -1/2)$$

Readers can refer [6] for how to derive the transform matrix $\mathbf{W}$ from the given filter. In implementation, we use fast reconstruction and decomposition operators derived from the filters, see [16] for more details.

Example 5.1 :(*Image Inpainting*) Our first application for our algorithm is image inpainting where only part of the information in the image domain are given. In this thesis, this means $\Lambda \neq \emptyset$ and $\Gamma = \emptyset$ in (1.1). We are only given part of data in the image domain. This is an interesting and important inverse problem. It arises in removing scratches in photos, in restoring ancient drawings, and in filling in the missing pixels of images transmitted through a noisy channel, etc. See [6, 16] for more on image inpainting. In our application, the missing data are the white words in the tested images.

Figure 5.1 is the inpainting results for our algorithm (4.4) for the analysis based approach, (2.1) for the the balanced approach and its APG algorithm. We can see that better result comes back in a fewer iteration for algorithm (4.4). This comes from the common fact that the Bregman iteration will give back edges quickly.

Example 5.2 :(*Super Resolution With Multiple Sensors*) We are going to reconstruct $\mathbf{f}$ by taking its low-resolution images using $K(K = 2$ in our implementation) multiple sensors of the same resolution but with different subpixel

Figure 5.1: Inpainting in image domain for the 'cameraman' image. Columns (from left to right) are the observed corrupted image, the recovered image by the analysis based model(4.1), the recovered image by the balance approach model (1.3), the recovered image by the APG algotirhm (2.19) respectively. The PSNR value of the recovered images are 35.7742, 34.3899,36.7285, respectively. The corresponding number of iteration are 9,100,13, respectively.

displacements, i.e., we just know part of the information in the transform domain. This means the application is for the case $\Lambda = \emptyset$ and $\Gamma \neq \emptyset$ in (1.1). Readers can refer [5] on how to get the index set Γ from the low-resolution images and [3, 10] for more details about super-resolution with multiple sensors. The first column of the images in Figure 5.2 are the given data. The second column are the observed high-resolution images. The third ,fourth and fifth column are the reconstruction high-resolution images by analysis based approach, balance based approach and its APG algorithm. All the low-resolution images are given in the first row and only parts of the low-resolution images are given for the rest.

Example 5.3 :(*Super Resolution with Different Zooms*) In this application, we are given part of the data in the image domain which is the first image in Figure 5.3. This means $\Lambda \neq \emptyset$. In the transform domain, we use two different sensors to get two different data sets. By using 2×2 sensor, we will get a low-resolution image, then we choose appropriate index set to get part of this low-resolution image which is the second image in Figure 5.3. By using 4×4 sensor (2nd level associated with the 2×2 sensor), we get a low-resolution image which is the third image in Figure 5.3. The second and third images are the given data in the transform domain, i.e., $\Gamma \neq \emptyset$. See [5] for more details on how to get the index set Γ from the given data in the transform domain and [5, 29] for more details on

Figure 5.2: 2×2 sensors for the 'boat' image. Columns (from left to right) are the available low-resolution images, the observed high-resolution images, the reconstructed high-resolution images by the analysis based model(4.1), by the balance approach model (1.3), by the APG algotirhm (2.19) respectively. The PSNR values of the reconstructed image are 31.7281,28.0557 22.4243, respectively for algorithm (3.10) (analysis based approach), 29.2638,29.1752,24.5309,respectively for algorithm (2.1)(balanced approach) and 35.8150,34.2161,28.8958 respectively for the APG algorithm (2.19).

Figure 5.3: Reconstructed super-resolution images for 'boat' image. Columns (from left to right) are low-resolution image from 4×4 sensors, part of low-resolution image form 2×2 sensors, part of original image, the reconstructed high-resolution image by the model (4.1), by the model (1.3) and by the APG algorithm (2.19) respectively. The PSNR value is 25.7972 for the analysis based model(4.1), 24.9855 for the balance approach model and 24.3859 for the APG algorithm (2.19) (1.3)

super-resolution with different zooms.

Example 5.4 :(*Data Reconsturction From Normal Vectors*) Now we consider another application for our model. For an image, if only parts of the normal vectors are given where, i.e.,$(D_x u, D_y u)_\Gamma$ are known where Γ is the supported index set. We will see that this problem can be considered as data reconstruction in transform domain, i.e., $\Lambda = \emptyset$ and $\Gamma \neq \emptyset$. This is similar with the second example but different problem in practise. Readers can refer [27, 35] for more details on the data reconstruction from normal vector.

The differential operators can be calculated approximately by the following formula

$$\begin{cases} D_x u(i,j) \approx u(i+1,j) - u(i,j) \\ D_y u(i,j) \approx u(i,j+1) - u(i,j) \end{cases}$$

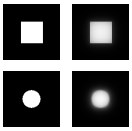


Figure 5.4: Image reconstruction from the normal vectors. The first column is the original images we used and the second column is the recovered images by (4.1) from the normal vectors of the boundary. The psnr are 28.9008,27.0168 respectively.

We use Haar wavelet in this implementation. Then by the four part of the wavelet coefficients (level one decomposition) can be calculated as follows

$$\begin{cases} LL(i,j) = \frac{\frac{u(i+1,j+1)+u(i,j+1)}{2}+\frac{u(i+1,j)+u(i,j)}{2}}{2} \\ LH(i,j) = \frac{\frac{u(i+1,j+1)-u(i,j+1)}{2}+\frac{u(i+1,j)-u(i,j)}{2}}{2} = \frac{D_x(i,j+1)+D_x(i,j)}{4} \approx \frac{D_x(i,j)}{2} \\ HL(i,j) = \frac{\frac{u(i+1,j+1)+u(i,j+1)}{2}-\frac{u(i+1,j)+u(i,j)}{2}}{2} = \frac{D_y(i+1,j)+D_y(i,j)}{4} \approx \frac{D_y(i,j)}{2} \\ HH(i,j) = \frac{\frac{u(i+1,j+1)-u(i,j+1)}{2}-\frac{u(i+1,j)-u(i,j)}{2}}{2} = \frac{D_x(i,j+1)-D_x(i,j)}{4} \approx 0 \end{cases}$$

Thus if we are given the data $(D_x u, D_y u)_\Gamma$, we can consider that we are given the corresponding wavelet coefficient of LH and HL on the index Γ. The readers can refer [4] for more information on this relationship.

The first image we tested in the implementation is a $0-1$ square, means the value in the square is 1 and 0 outsides. We take Γ to be the boundary of the square. The second tested image is a a $0-1$ disk, means the value in the circle is 1 and 0 outsides. We take Γ to be the boundary of the disk.

Chapter 6

Conclusion

In this thesis, we firstly give a review on the balanced approach two domain image restoration and its APG algorithm. The convergence rate of these two algorithm are also stated. Then we give a sufficient condition that ensures the exact recovery when the given data are $\mathbf{P}_\Lambda \mathbf{f}$ in the image domain and $\mathbf{P}_\Gamma \mathbf{W}\mathbf{f}$ in the transformed domain. The algorithm for the exact recovery is also proposed and we compared it with the balanced approach algorithm. We notice that the only difference between these two algorithms are the plunging denoising soft thresholding operator. Then the analysis based model is proposed and the correspondence algorithm are derived by using split Bregman method.

Bibliography

[1] A. Beck, and M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM Journal on Imaging Sciences, Vol. 2 , 1 (2009), 183-202.

[2] C. de Boor, R. DeVore, and A. Ron, On the construction of multivariate (pre)-wavelets, Constructive Approximation, Vol. 9, 2-3 (1993), pp. 123-166.

[3] N.Bose ,and K.Boo, High-resolution image reconstruction with multisensors, International Journal of Imaging Systems and Technology,Vol. 9, 4 (1998), 294-304

[4] Jianfeng Cai, Bin Dong, Stanley Osher, and Zuowei Shen, Image restoration: total variation, wavelet frames, and beyond, Journal of the American Mathematical Society, Vol. 25, 4 (2012), 1033-1089

[5] Jianfeng Cai, Raymond Chan, Lixing Shen, and Zuowei Shen, Simultaneously inpainting in image and transformed domains, Numerische Mathematik, Vol. 112, 4 (2009), 509-533.

[6] Jianfeng Cai, Raymond Chan, and Zuowei Shen, A framelet-based image inpainting algorithm, Applied and computational Harmonic Analysis, Vol.24, 2 (2008), 131-149.

[7] Jianfeng Cai, Raymond Chan, Lixin Shen, and Zuowei Shen, Convergence

analysis of tight framelet approach for missing data recovery, Advances in Computational Mathematics, Vol. 31, 1-3 (2009), 87-113

[8] Jianfeng Cai, Stanley Osher, and Zuowei Shen, Split Bregman methods and frame based image restoration, Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal, Vol. 8, 2 (2009), 337-369.

[9] A. Chai and Z. Shen, Deconvolution: A wavelet frame approach, Numerische Mathematik, Vol. 106, 4 (2007), 529-587.

[10] R.Chan, T.Chan, L.Shen, Z.Shen, Wavelet algorithms for high-resolution image reconstruction, Siam Journal on Scientific Computing, Vol.24, 4 (2003), 1408-1432

[11] R.H.Chan,L.Shen, and Z.Shen, A framelet-based image inpainting algorithm, Applied and Computational Harmonic Analysis, Vol.24, 2 (2008), 131-149.

[12] M.Ng, R.Chan, and W.Tang, A fast algorithm for deblurring models with Neumann boundary conditions, SIAM Journal on Scientific Computing, Vol. 21, 3 (2003), 851-866

[13] P.L. Combettes, and V.R. Wajs, Signal recovery by proximal forward-backward splitting, Multiscale Modeling and Simulation ,Vol. 4 , 4 (2006), 1168-1200.

[14] I. Daubechies, Ten Lectures on Wavelets, vol. 61 of CBMS Conference Series in Applied Mathematics, SIAM, Philadelphia, 1992.

[15] B. Dong, and Z. Shen, Pseudo-splines, wavelets and framelets, Applied and Computational Harmonic Analysis, Vol. 22, 1 (2007), 78-104.

[16] Bin Dong, and Zuowei Shen, MRA-based wavelet frames and applications, IAS/Park City Mathematics Series: The Mathematics of Image Processing, Vol 19, (2010), 7-158

[17] I. Daubechies, B. Han, A. Ron, and Z. Shen, Framelets: MRA-based constructions of wavelet frames,Applied and Computational Harmonic Analysis, Vol. 14, 1 (2003), 1-46.

[18] M.N. Do, and M.Vetterli, The contourlet transform: an efficient directional multiresolution image representation, IEEE Transactions on Image processing, Vol. 14, 12(2005), 2091-2106.

[19] D.L. Donoho, and P.B. Stark, Uncertainty principles and signal recovery, SIAM Journal on Applied Mathematics Vol. 49, 3(1989), 906-931

[20] E. Esser, Applications of lagrangian-based alternating direction methods and connections to split bregman, CAM report 9 (2009), 31.

[21] M.Fadili, J.-L. Starck, Sparse representations and bayesian image inpainting, SPARS'05, Vol. 1, Rennes, France, 2005

[22] M.Fadili, J.-L. Starck, and F.Murtagh, Inpainting and zooming using sparse representations, The computer Journal, Vol. 52, 1 (2009),64-79

[23] R. Glowinski, and P. Le Tallec, Augmented Lagrangian and operator-splitting methods in nonlinear mechanics, Society for Industrial Mathematics, 1989.

[24] S.S. Goh, and T.N.T. Goodman, Uncertainty principles in Banach spaces and signal recovery, Journal of Approximation Theory, Vol. 143, 1 (2006) 26-35.

[25] T. Goldstein, and S. Osher, The Split Bregman Method for L1-Regularized Problems, SIAM Journal on Imaging Sciences, Vol. 2 , 2 (2009), 323-343

[26] L.Borup, R.Gribonval, and M.Nielsen, Bi-framelet systems with few vanishing moments characterize Besov space, Applied and Computational Harmonic Analysis, Vol. 17, 1 (2004), 3-28

[27] Jooyoung Hahn, Chunlin Wu, and Xue-Cheng Tai, Augmented Lagrangian Method for Generalized TV-Stokes Model,Journal of Scientific Computing, Vol. 50,2 (2012), 235-264

[28] R. Jia, and Z. Shen, Multiresolution and wavelets, Proceedings of the Edinburgh Mathematical Society, Vol. 37, 2 (1994), 271-300

[29] M.V.Joshi, S.Chaudhuri, and R.Panuganti, Super-resolution imaging:use of zoom as a cue, Image and Vision computing, Vol. 22, 14 (2004),1185-1196

[30] A. Ron, and Z. Shen, Affine systems in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator, Journal of Functional Analysis, Vol. 148, 2 (1997), 408-447.

[31] Zuowei Shen, Kim-Chuan Toh, and Sangwoon Yun, An accelerated proximal gradient algorithm for frame-based image restoration via the balanced approach, SIAM Journal on Imaging Sciences , Vol. 4, 2 (2011), 573-596

[32] Z. Shen, K. C. Toh, and S. Yun, An accelerated proximal gradient algorithm for frame based image restorations via the balanced approach, SIAM Journal on Imaging Sciences, Vol. 4, 2 (2011), 573-596.

[33] X.C. Tai, and C. Wu, Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, Scale Space and Variational Methods in Computer Vision (2009), 502-513

[34] K.C. Toh ,and S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems, Pacific Journal of Optimization, Vol. 6, 20 (2010), 615-640.

[35] Tai-Pang Wu, Chi-Keung Tang, and Michael S. Brown, Heung-Yeung Shum, ShapePalettes: Interactive Normal Transfer via Sketching, ACM Transactions on Graphics, Vol. 26, 3 (2007), 44-48